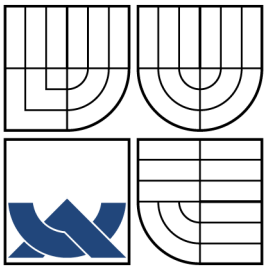
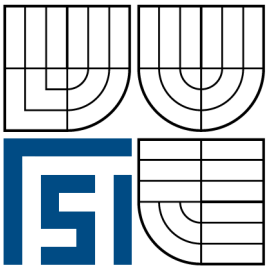


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

SLEDOVACÍ SYSTÉM ROZSÁHLÉ POČÍTAČOVÉ SÍTĚ
WIDE AREA NETWORK MONITORING SYSTEM

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR
AUTHOR
ZDENĚK BILL

VEDOUCÍ PRÁCE
SUPERVISOR
ING. PAVEL HERIBAN, PH.D.

BRNO 2008

LICENČNÍ SMLOUVA
POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení: **Zdeněk Bill**
Bytem: **U rybníka 415, 741 01 Nový Jičín**
Narozen/a (datum a místo): **13.12.1977, Nový Jičín**
(dále jen "autor")

a

2. Vysoké učení technické v Brně

Fakulta strojního inženýrství
se sídlem Technická 2896/2, 616 69 Brno
jejímž jménem jedná na základě písemného pověření děkanem fakulty:
doc. RNDr. Ing. Miloš Šeda, Ph.D.
(dále jen "nabyvatel")

Článek 1
Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):

☐ disertační práce
☒ diplomová práce
☐ bakalářská práce
☐ jiná práce, jejíž druh je specifikován jako .....
(dále jen VŠKP nebo dílo)

Název VŠKP:	**Sledovácí systém rozsáhlé počítačové sítě**
Vedoucí/ školitel VŠKP	**Ing. Pavel Heriban, Ph.D.**
Ústav:	**Ústav automatizace a informatiky**
Datum obhajoby VŠKP:	**4.11.2008**

VŠKP odevzdal autor nabyvateli v:

☒ tištěné formě – počet exemplářů **2**
☒ elektronické formě – počet exemplářů **2**

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2
Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užít, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizovaní výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti.

   ☐ ihned po uzavření této smlouvy
   ☐ 1 rok po uzavření této smlouvy
   ☐ 3 roky po uzavření této smlouvy
   ☒ 5 let po uzavření této smlouvy
   ☐ 10 let po uzavření této smlouvy
   (z důvodu utajení v něm obsažených informací)

4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3
Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísni a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: **17.10.2008**

.....	.....
Nabyvatel	Autor

Abstrakt

Práce se zabývá problematikou sledování rozsáhlé sítě s řádově až se stovkami aktivních prvků a tisícovkami uživatelů. Studie problematiky je situována v prostředí studentské počítačové sítě KolejNet, kde je v provozu informační systém. Informační systém v tuto chvílí obsahuje základy sledování této rozsáhlé sítě. Cílem je navrhnout systém, který umožní nepřetržitý monitoring aktivních prvků, mezi které patří sběr dat, důležitých událostí pro okamžité či pozdější vyhodnocení závažných údálostí, které mohou nastat.

Abstract

The subject of this work relates to monitoring of large networks consisting of hundreds of active elements and thousands of users. Study of the subject is situated in the Kolejnet computer network environment, where information system is being used. At the moment, the information system includes basic monitoring of this large network. The goal [of this study] is to architect a system that will allow continuous monitoring of active elements which includes collection of data and logging of important events for immediate and later evaluation of serious circumstances that can arise.

Klíčová slova

sledování sítí, wan, lan, systém, řízení, remote, snmp, mib

Keywords

monitoring networks, wan, lan, system, management, vzdálené, snmp, mib

Chtěl bych touto formou poděkovat vedoucímu diplomové práce Ing. Pavlu Heribanovi za metodické vedení, množství odborných konzultací a technických rad.

Obsah

1 Zadání

Cíle, kterých má být dosaženo:
Realizace systému pro sledování rozsáhlé počítačové sítě

Charakteristika problematiky úkolu:

- Návrh a realizace systému pro sledování rozsáhlé počítačové sítě
- Vyhodnocní náročnosti sběru dat v rozsáhlé síti se stovkami aktivních prvků.
- Vytvoření uzlů pro sběr a optimalizované ukládání dat z aktivních prvků sítě (informací o konfiguraci aktivních prvků, o aktuálním stavu a statistikách provozu jednotlivých portů)
- Vytvoření centrálního pracoviště pro zobrazování historií získaných dat, vyhodnocování potenciálních problémů a vzdálené ovládání aktivních prvků.

2 Úvod

V posledních letech zažil Internet v České republice podobně jako i ve světě bouřlivý rozvoj a zařadil se mezi nepostradatelné služby.

Uživatelé vyžadují vysokou dostupnost služeb a to za všech okolností s minimálními výpadky provozu. Důvodem je velké množství služeb, které Internet jako celek nabízí nejen jednotlivcům, ale i firmám a institucím. Ať jde o zábavu, vzdělání nebo bankovní služby, stal se Internet nepostradatelným pomocníkem.

Internet je tvořen v podstatě jen velkým množstvím menších sítí, které jsou vzájemně propojeny. Uživatel je připojen k Internetu v rámci své lokální sítě – LAN, případně rozsáhlé počítačové sítě – WAN.

Dostupnost služeb a bezpečnost uživatele je tedy výrazně omezena dostupností a bezpečností v rámci sítě, přes kterou je připojen. Současně s dostupností sílí tlak na bezpečnost a zabezpečení těchto sítí. Možným řešením je monitoring těchto sítí, který může zajistit včasné odhalení problémů a jejich rychlé vyřešení.

Práce je situována do prostředí studentské počítačové sítě KolejNet na VUT v Brně. Síť je budována od roku 2003 ve čtyřech areálech, které jsou propojeny optickou kabeláží. Koncoví uživatelé jsou připojeni výhradně metalickou kabeláží a převážná část uživatelů má svoji vlastní přípojku. Každý jednotlivý student by v budoucnu měl mít svou vlastní fyzickou přípojku. Tím se zamezí používání levných síťových prvků uživateli.

Vzhledem k poptávce studentů většina investic v minulých letech byla soustředěna do budování fyzické infrastruktury, jako jsou páteřní propoje, koncové přípojky.Další nemalá částka byla investována do páteřních a koncových prvků, které jsou plně managovatelné.

Počet současně připojených počítačů v síti již přesáhl v minulých letech opakovaně 6000. Síť takového rozsahu je provozována jen díky IS, který je průběžně vyvíjen a upravován, podle aktuálních potřeb, převážně studenty.

Cílem je zajistit nepřetržitý monitoring aktivních prvků sítě KolejNet a najít vhodné řešení. V budoucnu se počítá s využitím získaných dat pro provoz IS KolejNetu. Podaří se tak zajistit lepší kontrolu nad provozem sítě a zvýší se schopnost vyhodnotit vzniklé incidenty.

Do hardwaru a softwaru se investuje jen nezbytné množství prostředků. Na všech serverech jsou používány bezplatné operační systémy a aplikace a nepočítá se s nasazením drahých komerčních řešení.

3 Úvod do sledování

3.1 Přiblížení problematiky sledování

Sledování – monitoring sítě je činnost, při které opakovaně shromažďujeme informace o vybraných zařízeních, které jsou připojeny v rámci nějaké komunikační sítě.

Základními parametry monitoringu jsou:
- doba sledování
- předmět sledování
- způsob sledování

3.1.1 Monitoring podle doby trvání

Délka monitorování nemusí být vždy stejná a je dána typem informací, které se snažíme získat.

Rozdělení podle doby trvání:
- krátkodobý – je vhodné pro rychlé ověření parametrů chování
- dlouhodobý – je vhodné především pro získávání statistických dat

3.1.2 Výběr předmětu sledování

Rozlišujeme: náhodný cíl a cíl, který byl vybrán na základě nějaké indicie. Obvykle se jedná o prověření na základě oznámení. Subjekt provádí nestandardní činnosti, neobvyklé chování v síti apod.

Předměty sledování:
- náhodné
- cílené

3.1.3 Podle způsobu monitorování

Zde rozlišujeme podle účasti na monitorování:
- aktivní
- pasivní

Aktivní

Posíláme testovací data (datagramy, pakety ...), které po průchodu sítí vyhodnocujeme, nebo vyhodnocujeme reakci zařízení na testovaný vzorek.

Nevýhodou tohoto způsobu je například:
- přidaná zátěž sítě – ovlivnění běžného provozu
- obtížný způsob správné interpretace
- vypovídající hodnota výsledků

Příkladem takového monitorování je měření časového zpoždění, měření propustnosti "hrubou silou" nebo ztrátovost paketů.

Pasivní
Neposíláme testovací data, ale vyhodnocujeme časové a objemové charakteristiky provozu. Tento způsob monitorování neovlivňuje při správném použití provoz sítě. Získáváme tak cenné informace, které jsou aktivním přístupem nezjistitelné.

Z vlastností způsobu monitorování vyplývá použití aktivního a pasivního monitoringu. Aktivní jen vhodné pro dočasně a krátkodobě sledování, naopak pasivní je nasazován trvale a dlouhodobě.

Na základě stanovených parametrů můžeme informace: zpracovávat a vyhodnocovat. Případně činit na základě zpracování a vyhodnocení příslušné kroky.

3.2 Proč sledovat?

Prioritou provozovatele by měl být zájem poskytovat služby, které nabízí s určitými parametry a garancí. Protože současně s provozem sítí nabízí často přístup ke svým službám.

První důležitý krok je správný návrh sítě. Stěžejní částí je navrhnout správnou topologii, která bere v potaz dostupné technologie.

Ty se týkají, jak oblasti pasivních částí sítí (typ kabeláže, poloha uzlů apod.), tak aktivních částí (přepínače a směrovače). Návrh musí být učiněn s ohledem nejen na očekávanou zátěž, ale i s dostatečnou rezervou do budoucna.

Většina pasivních částí je limitována délkou a propustností. Největší omezení mají média založená na šíření signálu vzduchem, protože se jedná o velmi zarušené a těžko kontrolovatelné médium. Metalická média mají oproti tomu lepší parametry. Nejlepší parametry dosahují media optická MM, SM [1] – současně však roste pořizovací cena.

Maximální provozované rychlosti ve vzdálenosti 100 m se nyní pohybují pro vzduch mezi 100 Mbit/s a 300 Mbit/s. Pro metalické vedení na stejné vzdálenosti lze dosáhnout rychlosti v řádu jednotek Gbit/s. Nejlépe je na tom optická kabeláž, která zvládá bez problému až 1 Tbit/s (hlavně díky WDM[2]).

Bez dobrých znalosti vlastní topologie sítě nelze tyto poskytované služby garantovat. Proto je třeba znát vlastnosti vlastní sítě. Na základě topologie, lze pak snížit dopad výpadků. Pokud to není možně je nezbytné ji doplnit tak, aby při výpadku některého segmentu sítě zůstaly ostatní segmenty funkční.

Mezi nejdůležitější vlastnosti sítě patří:
- plánované
- provozní

Plánované vlastnosti jsou takové, se kterými bylo počítáno samotnou před stavbou sítě a zapojením infrastruktury. Zvažují se parametry požadované sítě, které by měly být při spuštění provozu,

[1] MM – multimode, SM – single mode
[2] Wavelength Division Multiplex – vlnový multiplex

3.3 Co sledovat?

Předmětem sledování by měly být parametry sítě, které chceme garantovat nebo udržet na určité kvalitativní úrovni.

Mezi základní parametry sítě patří:
- propustnost linek
- zpoždění

3.4 Jak a čím sledovat?

Důležitým faktorem je délka sledování a také na jakém principu je sledování založeno.

Rozdělení podle těchto kritérií:

krátkodobé – sledování je založeno především na protokolu ICMP[3]
dlouhodobé – monitoring probíhá pomocí protokolu SNMP
monitoring toku dat – dochází k monitoringu dat způsoby, které jsou uvedeny dále

Monitoring toku dat lze provádět pomocí:
- SMON[4] – vzdáleným monitoringem
- RMON [5]– monitorováním přepínačů (verze 1 a 2)
- Netflow[6] – monitoring IP toků firmy CISCO
- sFlow[7] – monitoring IP toků firmou InMon, RFC 3176

3.4.1 Krátký úvod do MIB

Definice [22]:

Informace o řízených objektech odděleny od funkčních modelů a jsou uloženy v bázi informací pro management. MIB[8] *a je to model řízených objektů (abstrakce systémových prostředků bez ohledu na jejich potřebě být řízeny), které jsou přístupné pro agenty a manipulovatelné manažery prostřednictvím protokolu managementu.*

MIB je hierarchicky uspořádaná množina objektů s definovanou syntaxí a sémantikou. Představuje specifikaci pro data (objekty), která musí agent (směrovač, počítač apod.) udržovat, a způsob přístupu k nim (čtení/zápis).

MIB se skládá ze dvou částí:
- textové – objekty jsou uspořádány do skupin
- modulu MIB – makro OBJECT-TYPE

MIB jsou nezávislé na řídícím protokolu včetně SNMP, ale požadavky kladené na SNMP se promítly při návrhu MIB. Díky tomu vznikl minimalistický model co se týče počtu řízených objektů. Jednoduchost a omezená velikost báze umožnily zaručit minimální dopad

[3]Internet Control Message Protocol – protokol řídících hlášení
[4]Switch Monitoring – monitoring switchů
[5]Remote Monitoring – vzdálené monitorování
[6]síťový protokol firmy CISCO
[7]standard pro sledování počítačových sítí
[8]Management Information Base

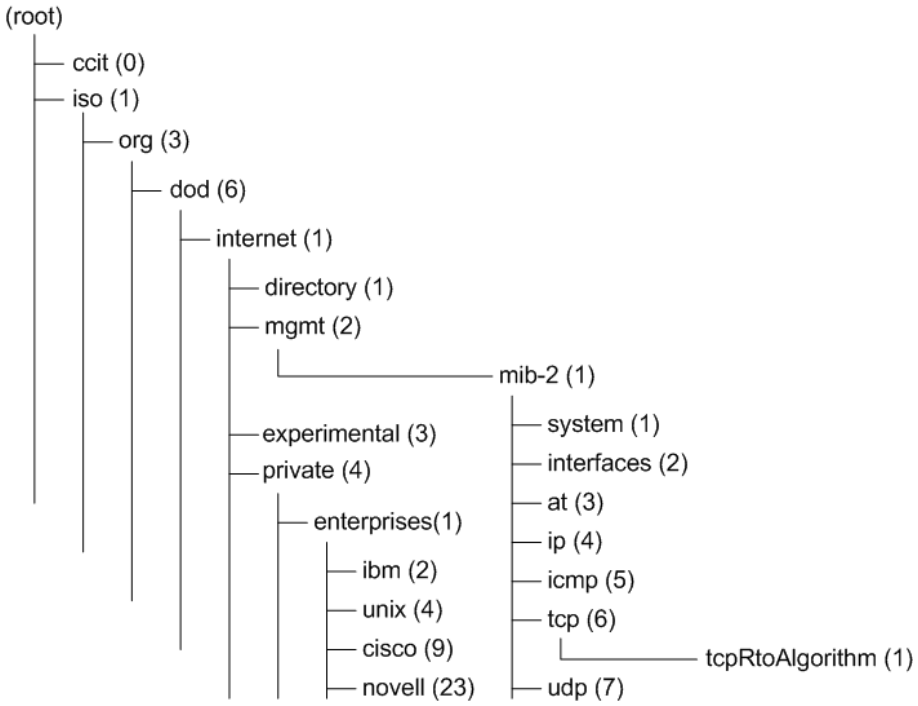
na činnost a složitost agentů, včetně paměťových kapacit.

MIB u SNMP se používá ve verzi II a rozděluje informace do 11 kategorií. Jejich seznam lze vidět v tabulce 3.1 .

Tab. 3.1 Kategorie MIB-II

system	operační systém agenta
interfaces	síťová rozhraní
addr-translation	Address translation, mapování adres
ip	informace o IP
icmp	informace o ICMP
tcp	informace o TCP
udp	informace o UDP
egp	informace o EGP
cmot	Common Management Information Protocol Over TCP/IP
transmission	informace o přenosových mediích
snmp	Informace o provozu SNMP

Každý typ objektu má:
- jméno
- identifikátor objektu – OI[9]
- textovým jménem popisem objektu – OD[10]



Obr. 3.1 Hiearchie identifikátorů objektů

[9]Object Identifier
[10]Object Descriptor

Ukázku hierarchie MIB pro představu lze vidět v tabulce 3.1.

Typy proměnných objektů:
- jednoduché – primitiva: integer, octet string, oi, null
- složené – tabulky: SEQUENCE OF primitiv
- definované – např. IpAddress jako OCTET STRING v délce 4 slabik

Typy řízených objektů:
- skalár - objekt má jednoho reprezentanta
- koncepční tabulka - vícenásobná reprezentace

3.4.2 ICMP

Je to protokol řídících hlášení a funguje na principu speciálních zpráv zasílaných tímto protokolem. V praxi se ICMP používá kvůli nespolehlivosti komunikačních spojů, nebo pokud je cílová stanice dočasně či trvale odpojena od sítě.

Tab. 3.2 Nejčastější typy ICMP zpráv

Typ	**Kód**	**Význam**
0	0	nedostupná cílová síť
3	1	nedostupná cílová stanice
3	2	nedostupný cílový protokol
3	3	nedostupný cílový port
3	4	datagram byl fragmentován
5	0	přesměrovat datagramy pro síť
5	1	přesměrovat datagramy pro stanici
5	2	přesměrovat datagramy pro síť a typ služby
5	3	přesměrovat datagramy pro stanici a typ služby
8	0	ping
11	0	překročena životnost datagramu

Také se používá v případě zahlcenosti linky a v případě vypršení životnosti datagramu. V tomto případě jsou informovány směrovače nebo koncová zařízení, které byly zdrojem datagramu.

ICMP využívá protokolu IP a datagramy se přenášejí podobně jako jiné datagramy. Nemají prioritní zacházení a proto není zaručeno jejich doručení. V praxi dochází často k jejich filtrování na zařízeních, které toto podporují a to v závislosti na jejich počtu nebo datovém toku. V tabulce 3.2 jsou uvedeny nejčastější typy ICMP zpráv.

3.4.3 SNMP

Patří mezi asynchronní, transakčně orientované protokoly (dotaz/odpověď).

Agent naslouchá a vysílá z portu 161. Manažer naslouchá na libovolném čísle portu a čeká na odpověď.

Existuje vyjimka pro trap, kde manažer naslouchá na portu 162.

Operace protokolu ve verzi 1:
- get-request – žádost o získání proměnné
- getnext-request – žádost o získání proměnné bez znalosti jejího jména
- get-response – odpověď na operaci žádosti
- set-response – nastav hodnotu proměnné
- trap – zpráva inicializovaná asychronně

Operace, které přibyly ve verzi 2:
- get-bulk – manažer může žádat o celou množinu informací
- inform – složí pro vzájemnou komunikaci manažerů

Verze 3 přináší nový formát zpráv, ochranu zpráv po cestě před: zničením, modifikací nebo odposlechem.

3.4.4 RMON

Jedná se o vzdálené monitorování, RMON MIB je soubor statistických, analytických a diagnostických skupin, které se zajišťuje pomocí sondprobes.

Sondy jsou agenti určení pro sběr informací přímo ze sítě. Pomáhají managementu se sběrem dat o provozu a to identifikací vytížených uzlů sítě a sběrem specifických paketů. Různé sondy spoléhají na různé zdroje informací: záznamy statistik, MIB zařízení apod. Mohou být součástí síťových zařízení nebo se jedná o samostatné počítačové systémy.

Mezi výhody patří možnost sběru dat i během výpadku komunikace s managementem, nebo zhoršených podmínek.

Zabudované sondy se používají při nepřetržitém sledování a mohou způsobovat snížení výkonnosti při vyšších rychlostech.

Samostatné sondy jsou poměrně drahé a vyžadují dobré plánování a možné je přenášet oproti zabudovaným sondám.

RMON nenahrazuje ani není alternativní vůči SNMP, ale tvoří doplňkový zdroj informací. Stejně tak RMON vidí pouze jeden segment LAN, tím se stávají další informace nedostupné zvláště v přepínaných sítích.

Bližší informace lze nalézt v literatuře [16].

3.4.5 SMON

Pracuje na základě sond v přepínačích, ty sbírají statistická data a vysílají poplašné zprávy a rozšiřuje tím RMON.

Umí navíc tyto funkce:
- zrcadlení – mirroring – kopíruje provoz z portu na port
- řízení – steering
- směrování provozu na vzdálený monitor

Jedná se zvlášťní skupinu sledování toku dat. SMON je protokol, který byl vytvořen firmou CISCO a analyzuje celý datový tok mezi vzdálenými uzly sítě. Narozdíl od jiných nástrojů se nezabývá jednotlivými přenášenými pakety a datagramy, ale zabývá se celým tokem.

V datovém toku se zajímá o tyto informace:
- typ protokolu
- zdrojovou adresu a port
- cílovou adresu a port
- typem služby

Provádí analýzu pro každý datový tok (doba trvání a počet přenesených dat) a vytváří pro něj samostatný záznam.

3.5 Management sítí

Je označován jako NMS (Network Management System) a je kombinací hardwaru a softwaru za účelem správy sítě.

Při propojení komunikačních sítí je nezbytné sledovat a řešit problémy s provozem. V rámci provozu komunikačních sítí dochází k zahajování, průběhu a ukončování nejrůznějších činností. Zaměřujeme se na monitoring očekávaného, tak nepředvídaného provozu.

Provoz v IP sítích lze sledovat na různých úrovních referenčního model ISO/OSI.

Management sítě v sobě zahrnuje:

- dohled
- kontrolu
- koordinaci činností
- správu síťových zařízení

Manažer sítě se může doptávat směrovače, stejně tak může směrovač informovat manažera zprávou o událostech, které nastaly.
Funkčnost managementu je podmíněná funkčností aplikační vrstvy a všech ostatních nižších vrstev.

Oblast managementu zahrnuje mnoho činností, které byly vykonávány od začátku implementace systému sítí. Ty nebyly zpočátku automatizovány, ale s rostoucím rozsahem, složitostí a heterogenitou sítí vzrostl tlak na úroveň nasazení automatizace a koordinace těchto činností.

Je třeba sledovat a vyhodnocovat činnost celého systému, aby mohlo být předcházeno změnou parametrů snížení efektivnosti sítě.

Základní otázky managementu:
- co se má monitorovat
- jakým způsobem monitorovat
- jak interpretovat výsledky
- využití nasbíraných údajů

Úkol managementu se rozděluje do těchto funkčních oblastí managementu:
- chyb – fault management
- konfigurace – configuration management
- účtování – accounting management
- výkonnosti – performance management
- bezpečnosti – security management

fault management
Patří mezi základní oblast, která zajišťuje funkce identifikace, izolace a odstranění poruch a problémů v činnosti, oznamování výstražných zpráv – *alerts*.

Alert zajišťuje informování uživatele o neobvyklých činnostech a poruchách. Management musí mít přístup v případě poruch k síťovým zařízením, aby bylo možné zjistit a vyhodnotit situaci.

configuration management
Úkolem je zjišťovat a dohlížet na stav sítě prvků a jejich konfiguraci (fyzickou či logickou), a případně ji měnit podle potřeby.

Konfigurace má zásadní vliv na:
- izolaci problémů
- změnu zátěže sítě

Management konfigurace je vhodné provádět přenosem zpráv datovým kanálem sítě, nebo využívat samostatného kanálu. Ten by měl zůstat aktivní pokud má datový kanál poruchu.

accounting management
Zajišťuje sběr a zpracování dat spojených s využíváním síťových prostředků. Typickým příkladem jsou uživatelé, u kterých probíhá sběr dat: využití sítě, délce trvání připojení, intezitě využití a počtu přenesených dat.

performace management
Umožňuje vyhodnocovat výkonnost systému sítě pomocí výkonnostních veličin.

Mezi sledované veličiny patří:
- propustnost
- odezva
- chybovost

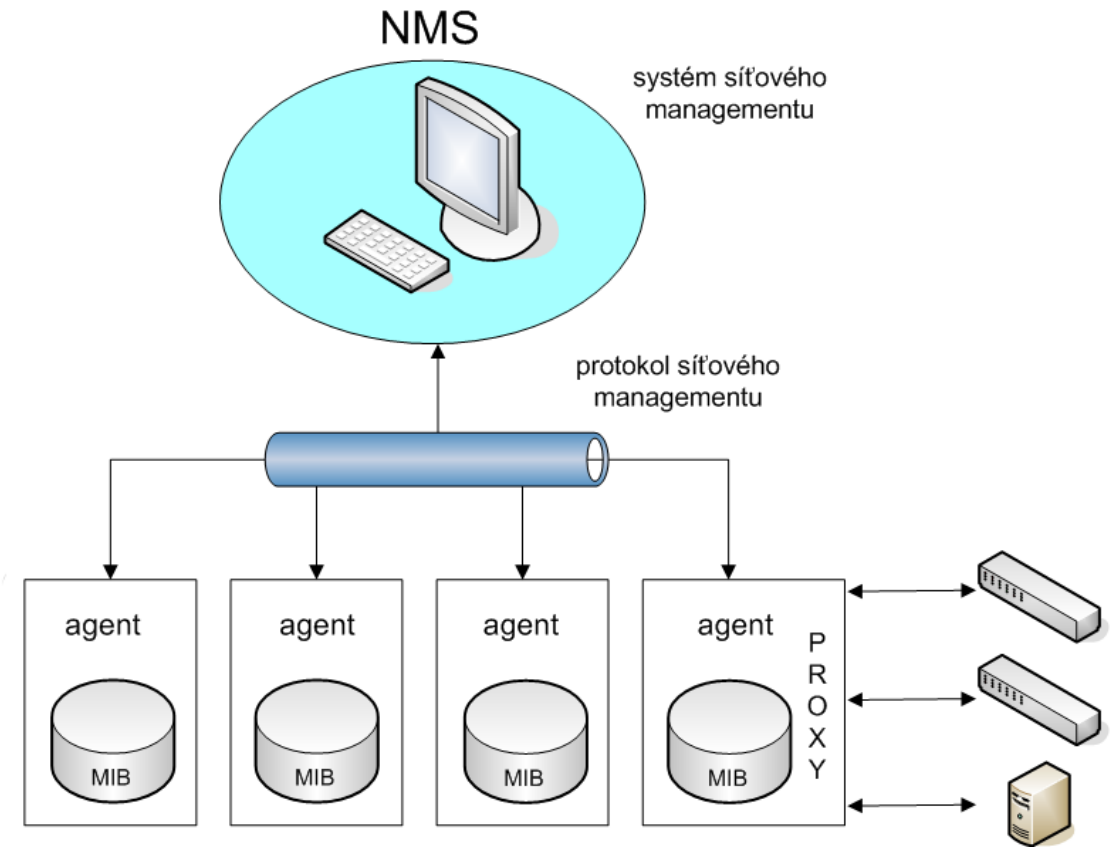
Získaná data lze použít jako podklad pro:
- změny v konfiguraci
- diagnostiku
- simulace provozu

security management
Zajišťuje bezpečnostní politiku mezi kterou patří autorizace pro přístup do sítě a k síťovým zařízením prostřednictvím autentizačních technik. Chrání síť před vnějšími útoky a zaznamenává neplatné pokusy pro získání přístupu k prostředkům sítě.

V rámci architektury TCP/IP je nejrozšířenějším řešením protokol SNMP, který zvítězil nad příliš složitým řešením SMIP/SMIS, které bylo definováno pro architekturu OSI.

3.5.1 Architektura, organizace a struktura managementu

Architektura managementu je postavena na dvou rolích: manažer a agent. Role jsou zobrazeny na obrázku 3.2 . Pro komunikaci mezi agentem a manažerem je na TCP/IP použit speciální protokol SNMP. Zde si objasníme jejich úlohu, princip fungování a náplň jejich činností.



Obr. 3.2 Model spolupráce manažer – agent

manažer
Jedná se o programové vybavení, které je umístěno na stanici síťového managementu NMS. Manažer řídí agenty přes objekty a řídí skupinu síťových zařízení, nad kterými provádí dohled.

Manažer může provádět centralizovaně následující úkony:

- monitoring
- kontrolu
- konfiguraci

Dále je zodpovědný za:
- sběr dat o stavu řízených objektů od agentů
- následnou reprezentaci sesbíraných dat
- management agentů prostřednictvím dotazů a příkazů
- delegaci agentů pro jednoduché řídící a monitorovací úkoly

agent
Programové vybavení na síťovém zařízení, které udržuje databázi objektů MIB – řídící informační bázi a komunikuje s manažerem. Agent vykonává čtení v bázi – MIB na základě odpovědí na dotazy od manažera. Dále vykonává modifikace nad objekty v bázi na základě příkazů ze strany manažera. Agent na základě nastavení může sám zasílat informace na základě nastalých situací a informovat tak manažera.

Na managementu řízení se mohou podílet také zařízení, která nemohou přijmout kompleletní software agenta a MIB, díky využití tzv. *zástupce agenta – proxy*. Ten je schopen spravovat software a MIB místo koncového zařízení a působí tak aktivně mezi managerem a řízeným zařízením.

4 Úvod do útoků

V této kapitole si ukážeme, jaké útoky lze provést na přepínaném ethernetu a jejich zneužití pro tvorbu DoS.

Závěrem se budeme zabývat zabezpečením síťových prvků a možnostmi, které v této době umožňují.

4.1 Útoky v ethernetu

V ethernetu lze rozdělit podle typu ethernetu a použitých prvků.

Typ prvku:
- nepřepínaný – je použit hub
- přepínaný – je použit switch

Hub – přijatá data rozesílá na všechny porty, lze tedy snadno odposlouhávat cizí komunikaci a útok je velice snadný. Nejedná se o zařízení, které může poskytnout bezpečný provoz.

Switch – přijatá data se podle MAC cílového počítače doručí na cílový port – je tedy bezpečnější, ale pomocí různých technik lze tuto bezpečnost narušit.

Odposlech cizí komunikace je nebezpečný jev a lze ho dále zneužít. Pro představu budou některé pokusy ilustrovány v Dos sekci.

4.1.1 Jak pracuje switch

Switch je zařízení pracující na linkové vrstvě → data adresuje podle MAC adres v hlavičce linkového rámce. Při příjmu dat nahlídne pro jakou MAC jsou určena a data pošle jen na port, kde je zařízení s touto MAC adresou.

Switch si drží informaci o tom, jaké MAC adresy se vyskytují na jednotlivých portech. Tyto informace jsou uloženy v CAM tabulce. Na jednom portu se může vyskytovat více než jedna MAC adresa. CAM [11] je tabulka obsahující relace MAC adresa, číslo portu, VLAN[12], čas do vypršení platnosti záznamu apod. Někdy se namísto CAM používá FDB[13].

Velikost a doba platnosti záznamů je dána výrobcem a typem switche.
Při startu switche je tabulka prázdná a při průchodu paketů se postupně nahlížením do paketů se plní. Ukládá se MAC odesílatele a port, ze kterého byl paket přijat. Následně je paket poslán podle záznamů v CAM na port na kterém je MAC, pokud záznam neexistuje, pošle se paket na všechny porty, kromě portu, ze kterého byl odeslán.

[11] Content-Addressable Memory
[12] Virtual Local Area Network
[13] Forwarding Database

4.1.2 Techniky v přepínaném ethernetu

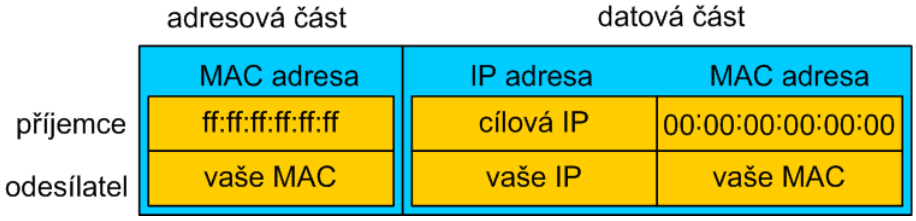
ARP Spoofing (Cache poisoning)
Jedná se o techniku u které je snahou modifikovat ARP cache switche. ARP je protokol, který se používá pro překlad IP adres na MAC adresy.

Switch pracuje na linkové vrstvě a adresuje pomocí MAC adres. K tomu používá paměť do které ukládá MAC adresy zařízení v síti a přiřazuje k nim porty, na kterých jsou umístěny.

Princip spočívá v tom, uchovat si svou MAC, ale současně předstírat adresu napadeného počítače. Je třeba tedy znalost své MAC adresy, která se dá zjistit snadno. Další podmínkou je znalost MAC adresy napadeného počítače.

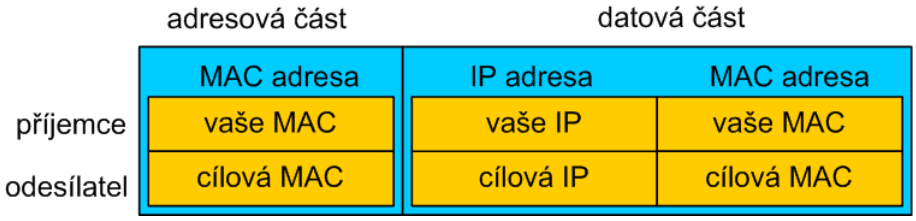
Zašle se tedy ARP Request, který používá pro adresování pouze MAC adresy. V datové části paketu se nalézají 4 položky pro adresy ARP Request a slouží pro to, aby se ozvalo zařízení, které má IP adresu příjemce uvedenou v datové části.

MAC adresa ff:ff:ff:ff:ff:ff je speciální tzv. *broadcast*. Pokud se zašle něco na broadcast, obdrží ho všechny počítače v *broadcast doméně*. Jak vypadá žádost o překlad je možné vidět na obrázku 4.1.



Obr. 4.1 Příklad žádosti o ARP překlad

Počítač, který tento paket dostane ho otevře, porovná IP adresu(cílovou) příjemce v datové části se svou. Pokud nastane shoda počítač odpoví, jak jde vidět na obrázku 4.2. Pokud shoda nenastane – paket zahodí.



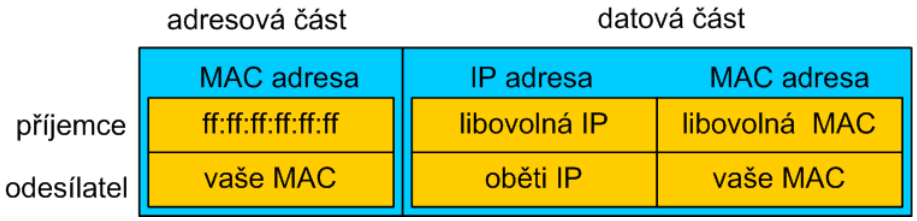
Obr. 4.2 Příklad žádosti o ARP Reply

Tento překlad se nedělá pokaždé, ale udělá se záznam do paměti – *ARP Cache*. Má určitou dobu platnosti a poté je záznam vymazán a překlad proběhne znovu.

Je jasné, že toto chování není bezpečné a je způsobeno stářím protokolu. V době návrhu se nehledělo příliš na bezpečnost.

Útok spočívá v tom, že oběti pošleme paket, ve kterém říkáme, že naše MAC je MAC brány. Bráně naopak pošleme paket, že naše MAC je MAC oběti. Tímto mechanismem docílíme toho, že obě zařízení budou provoz mezi sebou posílat na nás – útočníka. My můžeme v tuto chvíli data prohlédnout, modifikovat, vyplnit správné MAC adresy a zaslat. Dále je třeba zajistit, aby jsme v daných intervalech obnovovali špatné záznamy v ARP Cache.

Tohle byl příklad pro jeden počítač. Pro větší počet je třeba použít zmíněného broadcastu 4.3 – jedná se o tzv. *ARP Gratuitous Reply*.



Obr. 4.3 Příklad žádosti o ARP Gratuitous Reply

MAC Flooding

Tato technika spočívá v zaplnění CAM tabulky switche. Po zaplnění nemá prvek cílovou MAC v CAM tabulce a rozesílá jako hub.

Útok začíná vygenerováním takového paketů s rozdílnými MAC, aby došlo k zaplnění CAM tabulky. Lze tím nakazit i další switche v síti, kterým pak dojde také místo v CAM tabulce. Kapacita tabulek se pohybuje v od stovek až po statisíce.

Po zaplnění CAM není jasné, jak se bude switch chovat, záleží to od typu. Obecně se switch chová po takovém útoku jako hub. Data se pak rozešlou na všechny porty, kromě výchozího.

Lepší switche mají nastaven limit, který dokáže omezit počet MAC adres na jeden port. Ostatní si musí vystačit s tím, že jim stihnou vypršet včas záznamy než se CAM tabulka zaplní.

Rychlost generování paketů se na běžném PC pohybuje v řádů miliónů za minutu. Běžný čas pro vypršení záznamů se pohybuje od desítek sekund po jednotky minut.

Port stealing

Útok spočívá v kradení portů, toho dosáhneme tím, že switch aktualizuje CAM tabulku při přijmu paketu.

Jako u všch útoků potřebujeme zjistit MAC adresu oběti.
Následně budeme posílat pakety, které budou mít: cílovou adresu odpovídající naší MAC adrese a zdrojovou adresu nastavenu na MAC adresu oběti.

Switch po přijmutí paketu dojde k tomu, že oběť byla připojena na portu útočníka a upraví si na základě toho CAM tabulku. Protože cílová MAC adresa je na stejném portu, nepošle se paket dále.

Pokud je oběť na jiném switchi, tak použijeme cílovou adresu na broadcastovou adresu. Tím si zajistíme, aby pakety pro oběť byly poslány nám.

Paket můžeme prohlédnout (případně modifikovat), pro zaslaní oběti musíme opravit CAM tabulku.

Opravu provedeme tak, že přestaneme posílat pakety pro ukradení portu a pošleme *ARP Request*. Obět odpoví *ARP Reply*, paket se dostane na switch, podívá se do pakety a opraví záznam v CAM tabulce.

Tím, že obdržíme *ARP Reply* máme jistotu, že záznam v CAM byl opraven a zachycený paket pošleme oběti. Pak celý proces podle potřeby opakujeme.

Tento útok je obtížně vysledovatelný a lze ho odhalit jen: zvýšenou aktivitou na switchi, vypsáním CAM tabulky nebo velkým počtem ARP dotazů.

DHCP Spoofing

Při tomto útoku dochází k využití faktu, že v jedné síti může běžet několik DHCP serverů. Dojde tedy útočníkem k zprovoznění DHCP[14] serveru a podstrčením falešných údajů. Mezi tyto údaje patří třeba gateway nebo DNS server.

gateway

Nebo-li brána, je označováno zařízení, které zajišťuje spojení lokální sítě s jinou sítí. Pokud počítač potřebuje komunikovat s jiným počítačem vezme IP adresu a masku podsítě v binárním tvaru a provede logické vynásobení. Tuto operaci provede se zdrojovou, tak cílovou IP adresou.

Vynásobením získáme adresu sítě. Pokud se obě sítě shodují, je počítač ve stejné síti. Pokud ne, počítač je v jiné síti a data jsou poslána přes bránu.

Oběť při připojení do sítě zasílá do sítě broadcastový paket *DHCP Discover*, tím si zažádá o přidělení parametrů DHCP serverem. DHCP server odpovídá pomocí DHCP Offer, ve kterém mu zašle parametry. Do této chvíle je ještě vše v pořádku.

Následuje odpověď všech DHCP serverů, které obdržely žádost. Tady platí obvykle pravidlo nejrychlejšího – klient přijme parametry a odpoví nejrychlejšímu serveru paketem *DHCP Request*. Následně mu sdělí, že by tyto parametry přijal. Server mu poté pošle DHCP Ack a potvrdí dohodnuté parametry.

Pokud oběť je již připojena probíhá útok vyčerpáním volných IP adres, které server přiřazuje. Následně klient po vypršení času, po kterou mohl parametry využívat, zasílá *DHCP Discover* a chovají se jako by v síti nebyly.

ICMP Redirect

Útok využívá ICMP zprávy typu 5. Jedná se o zprávy, kterými se optimalizuje routování dat sítí. Využívá se podtypu *přesměrování na hostitele*.

Princip je následující: pokud na bránu přijdou data a ona zjistí, že je výhodnější posílat přes jinou bránu, tak o tom podá zprávu právě pomocí ICMP. Počítač obdrží tuto informaci a upraví si směrovací tabulku.

Útok začne posláním ICMP Redirect a přesměrujeme datový tok oběti na útočníka.

[14] protokol dynamického přidělování síťových parametrů: IP adresa, maska, DNS servery apod.

DNS Spoofing

Jedná se o kategorii útoků, která spočívá v povržení IP adresy v paketu, který se vrací jako žádost o překlad doménového jména na IP adresu.

Způsobů provedení je celá řada a lze je využít i pro útoky mimo lokální síť. Dá se využít k útoku na jednoho tak tisíce uživatelů. Toto téma je obsáhlé, že případné zájemce lze odkázat na vyhledávač, protože útoků je velké množství a pravidelně přibývají nové varianty.

4.2 DoS

DoS je zkratka pro *Denial of Service*, což je v překladu odmítnutí služby. Prakticky se takto označují útoky, které se snaží o znepřístupnění určitý služeb.

Jedním z důvodů monitoringu je možnost vyhodnocení v průběhu odkud útok přichází a zastavení ho, a nebo k provedení zpětné analýzy.

4.2.1 Příčiny útoků

Příčin DoS útoků je celá řada. Hlavní příčinou je motivace vyřadit, nebo zásadně omezit konkrétní službu.

Aby mohl být takový útok úspěšný, je třeba znát cíl a jeho vlastnosti.

Mezi takové informace patří:
- OS provozovaný na počítači, který chceme napadnout
- služba, kterou chceme napadnout
- šířka pásma, kterou disponuje služba
- zranitelná chyba, která bude využita

Jednou z hlavních příčin DoS útoků, je snaha využití chyb OS a jejich služeb, nebo aplikací a zneužít v co nejkratší době co nejvíce PC.

Takové PC může být pak využito k různým nepříjemným praktikám pro napadeného:
- šíření materiálů chráněných autorským zákonem
- skrytí vlastní identity
- skrytí vlastní identity za účelem dalších útoků
- znefunkčnění OS

Rozdělení podle počtu útočníků:
- DoS Je to typ útoku, při kterém musíme mít přístup k počítači na který chceme útočit
- DDoS Tento útok můžeme provést, jak napovídá název na vzdálený počítač

DoS útoky převažovaly hlavně v minulosti v dnešní době je využíván DDoS, což je vlastně podmnožina DoS a jedná se o distribuovaný útok.

4.2.2 Dělení DoS útoků

DoS můžeme dělit podle různých kritérií, pro příklad si uvedeme některé z nich.

Podle přístupu:
- lokální – je to typ útoku, při kterém musíme mít přístup k počítači na který chceme útočit.
- vzdálené – tento útok můžeme provést na vzdálený počítač.

4.2.3 Kategorie DoS útoků

Distribuované

To v praxi znamená, že útok nejde z jednoho počítače, ale z několika počítačů, jejich počet může být ve stovkách až třeba v tisících.

Flood

Záplavové útoky patří mezi nejjednodušší útoky, které spočívají ve vygenerování co největšího toku nebo počtu paketů. Cílem je zahltit linku oběti. Záměrně je uveden datový tok a počet paketů, protože se jedná o odlišný způsob.
Ohroženy jsou hlavně linky s menší propustností, ale není problém zahltit cíl, který má i velmi rychlé spojení. Nebezpečnost spočívá v tom, že jim nelze v podstatě účinně zabránit, zvláště v kombinaci s DDoS. Jediná obrana je pomoc poskytovatele, který daný provoz může odfiltrovat.

Využívající chyb a vyčerpání systémových prostředků

Útoky této kategorie využívají zranitelnosti softwaru nebo hardwaru. Nejčastější chyby jsou v návrhu systému nebo v implementaci.
Podstatná je znalost chyby a nalezení vhodného cíle, nebo naopak znalost cíle a hledání známe chyby, která umožňuje tento typ útoku.
Velkým nebezpečím je pokud není oprava chyby instalována včas nebo vůbec.

MITM [15]

Tento typ znamená v překladu "muž uprostřed", a jak název napovídá jedná se o útoky, které zneužívají pozice v komunikaci – tedy útočník zneužívá své pozice v komunikaci.

Lze ho rozdělit na dva druhy:
- lokální
- globální

Reflektivní a zesilující

Útoky dovolují zahltit kapacitu linky oběti, která má menší propustnost, než linka útočníka.

Reflektivní
Snaží se zahltit za pomocí jiných počítačů (routerů).

[15] Man in the middle

Zesilující
Útočník pošle data o určitém objemu a pomocí reflektivního útoku dojde k zesílení.

Nechtěné
Jak již naznačuje název jde o útoky, které nebyly vyvolány úmyslně a neexistuje útočník v pravém slova smyslu.

4.2.4 Bližší seznámení s jednotlivými kategoriemi

Flood

Zde jsou uvedeny bližší informace o jednotlivých typech.

- ICMP – pro útok se vyžívá ICMP protokolu a používají se pakety typu ICMP Echo. Nebezpečí spočívá v tom, že ICMP je obvykle krátký paket, ale lze poslat podle specifikace paket o velikosti až 65 535 B. V případě Echo je poslán paket a zpět je posláno Reply a to o stejné velikosti. Další úskalí je ve snadnosti útoku, lze využít k útoku běžný příkaz ping a zvládne ho i začátečník.

- UDP – je zde využita bezstavovost UDP protokolu – nenavazuje se spojení jako u TCP. Využívá se služby *echo*, kdy zaslaná data na její port pošlou nazpět. Při podvržení zdrojové IP adresy lze snadno útočit na námi určený cíl. S využitím více počítačů, lze pak snadno zajistit DDoS na určený cíl.

- TCP – jsou založeny na TCP. Setkáváme se s pojmenováním podle příznaků, které jsou použity: SYN Flood, ACK Flood, RST Flood, FIN Flood, UGR Flood, PSH Flood. V podstatě až na SYN a RST se jedná o ten stejný útok a cíl je stejný – vygenerovat patřičný datový tok a zahltit linku.
  SYN Flood je jiný v tom, že se jedná o útok, který vyčerpává systémové prostředky.
  RST Flood je využíván k resetování spojení.

Využívající chyb a vyčerpání systémových prostředků

Samotný útok probíhá následovně: zaslání velkého počtu paketů, které bývají zpravidla speciálně upraveny a to tak, aby došlo k vyčerpání systémových prostředků.

Mezi ně patří zejména vytížení CPU, nebo aplikace začne konzumovat více paměti. Následkem toho dojde k vyčerpání volné paměti a systém začne používat odkládací oddíl [16].

Typy útoků:
- Ping of death – zneužívá chyby v implementaci ICMP, týká se starších OS.
- Teardrop – podobný PoD, založen na zneužití fragmentace paketů, špatná implementace pod OS.
- SYN Flood – zneužívá chyby v implementaci TCP handshake.
- RPC Named Pipes – zneužívá chyby v Remote Procecedure Call.
- Stream a Raped – využívá špatné implemetace zpracování vadných paketů.

[16] swap space

- Land – oběti se posílají zfalšované pakety, kde je podvržena adresa na adresu oběti, za jistých okolností se dá obejít firewall.

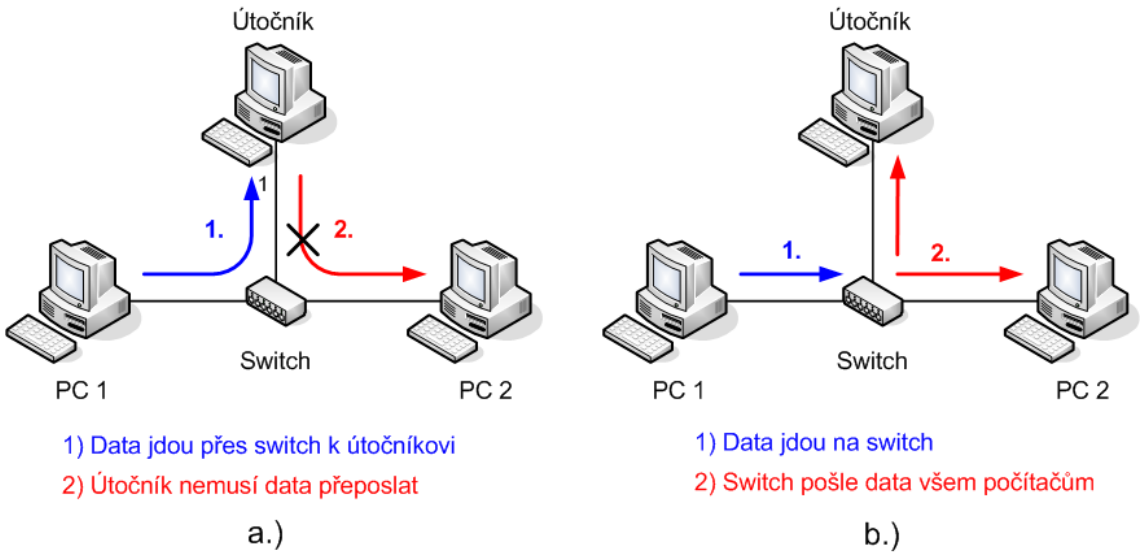
Toto je jen část útoků, existuje řada dalších, zvláště těch co využívají chyb OS, ale ty jsou v rámci řešeného úkolu nepodstatné.

Tím dojde k zahlcení systému, který nestíhá zpracovávat běžný provoz.

MITM

Aby mohl útočník zneužít svého "postavení" v síti, musí se stát uzlem, přes který prochází komunikace.

Toho lze docílit různými způsoby a my si uvedeme jen techniky v lokální síti: ARP Cache poisoning, DHCP Spoofing, ICMP Redirecting, DNS Spoofing, Port stealing, DNS Spoofing.



Obr. 4.4 Man in the middle: a.) Ostatní útoky b. MAC flooding

Naproti tomu MAC Flooding se použít nedá, protože nelze zabránit tomu, aby nebyly data doručena koncovým uzlům, i když jsme schopni odposlouchávat komunikaci v segmentu.

Objasnění problematiky je na obrázku 4.4.

Reflektivní a zesilující

- DNS zesilující – zneužívá chyby v DNS, zesílení je dosaženo pomocí pokládání dotazů více DNS serverů a podvržení IP adresy
- Smurf – podobný ICMP flood, ale využívá se zesílení, protože se posílá echo namísto na IP adresu oběti na IP adresu sítě
- Fragle – využívá se posílání dat na adresu sítě, využívá se UDP a služeb echo a chargen

- SYN flood – využití více počítačů a podvržení IP
- TTL flood – využívá posílání paketů se zvyšujícím se TTL[17], zesílení je malé

Nechtěné

Jeden z možných příkladů: na hodně navštěvovaném webu (provoz je zajištěn několika servery) je odkaz na zdroj – server, který není na takovou zátěž připraven.

V lokální síti se lze sekat s nechtěným útokem způsobeným vadnou síťovou kartou, nebo drivery. Jistou dobu s tímto měly problémy síťové karty od firmy nVidia.

Tyto útoky se projevují generováním datagramů nebo paketů, u kterým může být vadné třeba CRC, dojde tak k přetížení switche.

Další příkladem z praxe je multicastový provoz, který má datový tok překračující nejmenší rychlost rozhraní switche. Takže v sítí, kde je šířen multicastový provoz např. 20 Mbit/s, po připojení síťového zařízení rychlostí 10Mbit/s a požádání multicastového provozu dojde k zahlcení rozhraní.

Současně s tím dojde i k problémům na samotném switchi, který si s touto situací neumí poradit. Samozřejmě záleží na implementaci podle jednotlivých typů a výrobců.

4.2.5 Zabezpečení na prvcích

Uvedli jsme si některé známé typy útoků, zde bude uveden seznam opatření, který může některým problémům předcházet, nebo je eliminovat.

Zabezpečení přístupu k managementu

Při správě prvků s managementem je třeba zajistit, aby byl přístup k prvkům dobře zabezpečen. Jinak hrozí, že kontrolu nad správou sítě převezmou nepovolané osoby. V nezabezpečené síti se nedoporučuje používat nezabezpečené služby.

Běžné přístupy k managementu těmito protokoly:
1. http
2. https
3. telnet
4. ssh
5. snmp

Nezabezpečené služby:
1. http
2. telnet
3. snmp (verze 1, 2, 2c)

Tyto protokoly nejsou považovány za bezpečné a není je vhodné používat ve veřejných sítích.

[17] Time to live

Zabezpečené služby:
1. https
2. ssh
3. snmp (verze 3)

Samozřejmě služby jsou bezpečné jen při dodržení pravidel pro práci s nimi. Jedná se zejména o práce s šifrovacími klíči a ověřováním protistrany.

Zabezpečení uživatelských portů

Tyto omezení dnes podporuje drtivá většina prvků mající management. Následující omezení podstatně zvyšují bezpečnost samotné sítě a mohou zamezit výpadkům nebo omezení provozu sítě.

omezení počtu MAC adres na portu
Je to omezení na počet MAC adres připojených k portu. Při dosáhnutí limitu není dalším MAC adresám umožněna komunikace.

omezení na oprávněné MAC adresy
Na portu je seznam oprávněných MAC adres a jen vyjmenovaným je umožněn přístup do sítě. Toto omezení je silně restriktivní a vhodné např. pro podnikové sítě.

DHCP snooping
Jedná se rozdělení portů na důvěryhodné a nedůvěryhodné. U důvěryhodných portů není omezen provoz DHCP a může být na něm provozován DHCP server. Port je označen jako nedůvěryhodný a tak prvek nepřijímá *DHCP Offer*, tím pádem není z portu umožněno provozování DHCP serveru.

5 Místní situace

5.1 Stručná charakteristika sítě

Definice a účel sítě je uveden v pravidlech provozu v Pravidlech provozu počítačové sítě KolejNet [1]:

KolejNet je počítačová síť budovaná v areálech kolejí VUT v Brně (dále jen "VUT") pro potřeby studentů. Skládá se z přípojek v počítačových učebnách (Kolejní 2 a Purkyňova 93) a z přípojek na pokojích studentů. Prostřednictvím Brněnské akademické počítačové sítě (BAPS) je připojena na národní síť pro vědu, výzkum a vzdělávání (CESNET 2) a jejím prostřednictvím do sítě Internet.

Síť KolejNet je provozována v těchto lokalitách:

- Kolejní 2
- Purkyňova 93
- Kounicova 48
- Mánesova 12

Lokality jednotlivých areálů jsou propojeny optickou kabeláží, která je spravována CVIS[18].

Všechny areály jsou v kruhové topologii a mají k dispozici šířku pásma 10Gbit/s. Vyjimku tvoří areál Mánesových kolejí, který je připojen rychlostí 1 Gbit/s (důvodem je malý počet uživatelů).

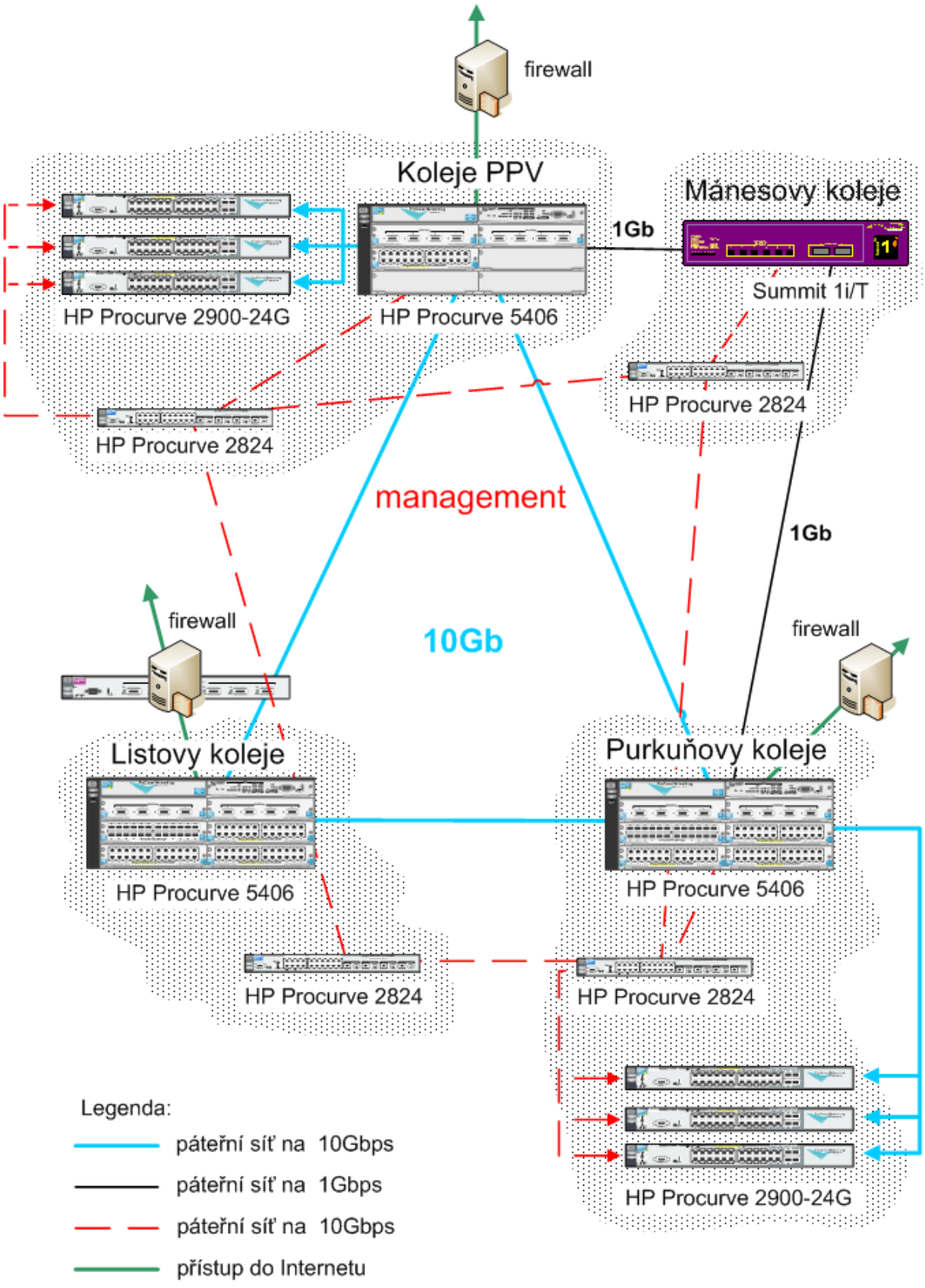
Všechny areály jsou propojeny klasickou páteřní sítí, která zajišťujě běžný provoz na rychlosti 10 Gbit/s. Souběžně s klasickou sítí je použita tzv. *management síť*, která je jak logicky, tak fyzicky oddělena od obvyklého provozu.

Management sítě slouží převážně pro správu páteřních prvků a další zařízení, který při problému s klasickou částí sítě umožňuje zjistit lokalitu a typ problému, který výpadek způsobil. Výhodou management sítě je tedy nezávislost na provozu v uživatelské části sítě, zvýšená bezpečnost proti výpadkům a zvyšuje se bezpečnost správy prvků, které jsou konfigurovány tak, aby komunikovaly co se správy týče jen z této části sítě.

Páteř sítě KolejNet je tvořená převážně spoji o rychlosti 10 Gbit/s. Provoz je zajištěn centrálními přepínači *HP Procurve 5406* a jedním *Summitem 1i/T*, který je připojen rychlostí 1 Gbit/s.

Tyto prvky tvoří hlavní páteř mezi areály. Lokální páteř v rámci areálu zajišťují switche HP 2900-24G, které zajišťují vysokou propustnost až 10 Gbit/s. V budovách se pak dále používají switche HP 2524, HP 2650 a vyjimečně HP 2824.

[18] Centrum výpočetních a informačních služeb



Obr. 5.1 Páteřní síť KolejNetu

Pro představu uvedeme nejdůležitější informace o jednotlivých areálech v tabulce 5.1

Tab. 5.1 Souhrné informace o areálech

Areál	Koleje PPV	Purkyňovy koleje	Listovy Koleje	Mánesovy koleje
páteřních uzlů	4	2	1	1
koncových uzlů	19	8	4	1
přípojek	2836	1985	1056	264
serverových uzlů	1	2	1	0
serverů	10	8	5	1

Celkem je tedy provozováno na přes 6100 uživatelských přípojek, které jsou během školního roku aktivně používány. Minulý rok bylo registrováno přes 6300 připojených počítačů.

Drtivá většina pokojů je vybavena počtem přípojek, které odpovídají počtu lůžek na pokoji. Existují ovšem vyjimky, kde je menší počet přípojek než lůžek a studenti používají vlastní rozbočovač. Také přibývají případy, kdy jeden student současně provozuje více, jak jeden počítač – obvykle desktop a notebook.

5.2 Poskytované služby

Provoz sítě je nepřetržitý 365 dní v roce, 7 dní v týdnu, 24 hodin denně. Částečné výpadky jsou možné jen během prázdnin. Probíhají výstavby dalších úseků sítě, modernizace, změny v konfiguraci sítě a další nezbytné úpravy.

V rámci sítě je přístup k Internetu, tak k doplňkovým službám, které jsou provozovány. Správu sítě umožňuje IS, který patří k základním nástrojům pro správu a údržbu.

Tab. 5.2 Servery KolejNetu

server	**operační systém**	**hardware**	**dbs**
jaja	FreeBSD 4.x	iP4, SCSI	
jaja2	FreeBSD 4.x	iP4, IDE	PostgreSQL
snake	FreeBSD 4.x	iP4, IDE	
fenix	FreeBSD 6.x	iP4, IDE	MySQL
dogs	FreeBSD 6.x	amd64, SCSI	MySQL
blade	FreeBSD 4.x	iP4, IDE	MySQL
arestroje	FreeBSD 4.x	iP4, IDE	
edna	Linux	iP4, IDE	
bedna	Linux	iP4, IDE	
anika	Linux	iP4, IDE	

Současně se provozují běžné služby DHCP, DNS, SNMP, NTP, NET-TV[19] apod., které jsou provozovány na řadě serveru rozmístěných na různých areálech.

Servery běží na různých OS systémech, základní přehled je v následující tabulce 5.2.

Kritické služby jako jsou DHCP a DNS obstarává v každé budově samostatný server. Tyto služby jsou provozovány redundantně. Na každou lokalitu připadají čtyři DHCP servery, které se navzájem zálohují.

DNS je provozována na primárním a sekundárním serveru, cache servery jsou provozovány v každé budově.

[19] jedná se o šíření TV signálu pomocí multicastového streamu

6 Řešení

6.1 Hardware

Servery KolejNetu jsou historicky založeny na platformě PC. Tato platforma umožnila rychle a levně nasadit potřebné služby pro provoz sítě. V dnešní době se pomalu přechází k profesionálnímu řešení serveru od světoznámých značek HP, DELL apod. a to z důvodu rychlé dostupnosti servisu.

K řešení byl vybrán server DELL PowerEdge2950.

6.1.1 Výběrová kritéria serveru

Podpora a servis

Jsou nezbytné pro provozování samotného serveru pod požadovaným OS.
Pokud by šlo o neznačkový server, je možné např. řadič vybírat obvykle svobodně bez ohledu na výrobce HW – pouze s přihlédnutím na aktuální konfiguraci (omezeno obvykle jen sběrnicí serveru).

Značkové firmy mají časté kritérium pro poskytnutí záruky a servisu – provozování HW jen dodaného výrobcem serveru.

Toto omezení má své opodstatnění, ale současně značně zužuje výběr produktů a znemožňuje například vybrat libovolný řadič disků.
Vzhledem k nejčastěji provozovanému OS na KolejNetu, byl kladen důraz na podporu pod OS FreeBSD (Linux je dnes obvykle podporován standardně). Server byl tedy vybrán především na základě tohoto kriteria.

Rychlost a výkon

Požadavek na rychlost a výkon byly v minulosti velmi důležité kritérium hlavně u CPU, RAM a rychlost sběrnice, ale mezi další zařízení patří i diskový subsystém a síťová karta.

CPU – dneska dosahují vysokých frekvencí a výkon lze navyšovat používáním více procesorů(jader). Vyjimkou dnes není ani 4, 6 či 8-mi jádrový procesor. Při výběru CPU je výhodné použít raději menší počet jader na vyšší frekvenci, než větší počet procesorů na nižší frekvenci. Výběr samozřejmě záleží na nejčastějším druhu úlohy, která bude zpracovávana.

RAM a sběrnice – vhodné je vybírat vyšší frekvenci pamětí a sběrnice. Např. dualchannel paměti neposkytují slibovaný nárůst propustnosti – záleží na chipsetu, ale lze jednoduše ověřit utilitou *memtest* . Velmi důležité je mít dostatek operační paměti, aby OS neswapoval, protože dochází k zásadní degradaci výkonu serveru.

diskový subsystém – důležitý faktor je přístupová doba, přenosová rychlost (samotného disku, tak i sběrnice) a MTBF. Pokud budeme používat RAID je limitem řadič – maximální rychlost sběrnice.

NIC – u síťových karet nás zajímá jejich maximální přenosová rychlost (100Mbit/s, 1Gbit/s, 10GBit/s), ale také zpoždění (v nanosekundách).

Další možnosti

Mezi sledované možnosti patří zejména zprovoznění hardwarového RAID, redundantní napájení, hotswap disky apod. Dokáží v praxi snížit náročnost servisních zásahů a zvýšit dostupnost serveru a tedy i služeb.

Rozšiřitelnost

Během životnosti serveru lze dostatečně rozšířit diskovou kapacitu, velikost RAM, výměna CPU za rychlejší a podobně.

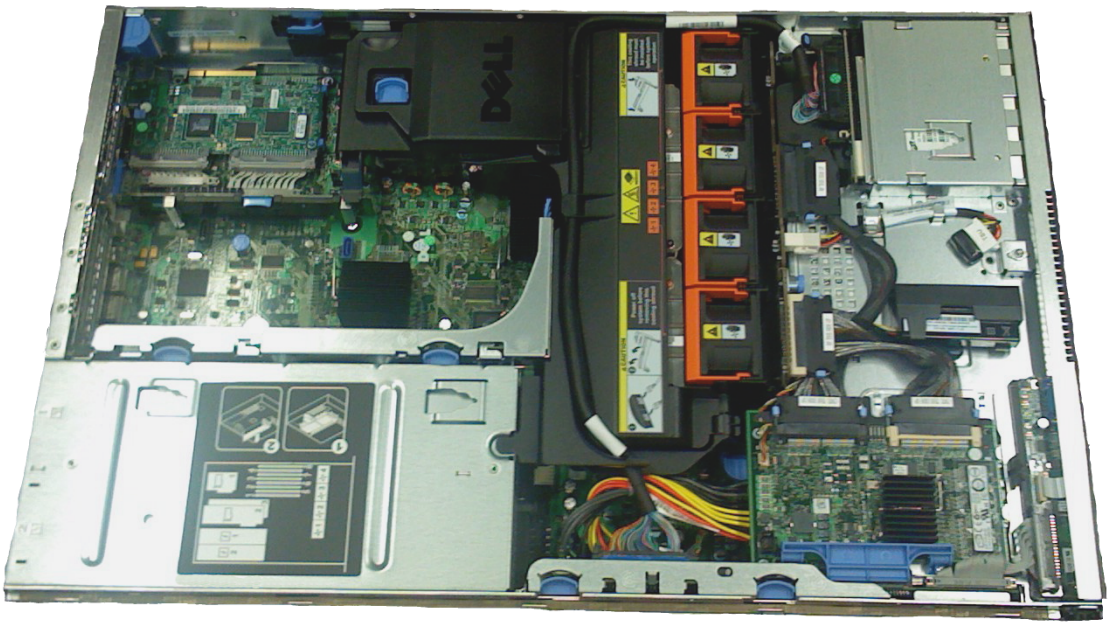
6.1.2 Konfigurace serveru

Server je postaven na platformě PC, kompletně dodaný výrobcem s touto konfigurací:

- *provedení – rack, 2U*
  vybráno provedení vhodné do racku, kde bude server umístěn

- zdroj – 2×, výkon 2x750W
  z důvodu možného výpadku jednoho zdroje a tím pádu serveru byla vybrána varianta se dvěma zdroji, které se vzájemně zálohují

- *procesor: 2× Intel (R) Xeon (R) 5150*
  celkem instalován 4 jádra s taktovací frekvencí 2.66GHz, 4MB (společná pro 2 jádra) a 1333MHz sběrnicí
  server byl osazen maximálním počtem jader s dostatečnou frekvencí



Obr. 6.1 Server DELL 2950

- *RAM – 2GB (4× 667 DIMM)*
  maximum paměti, kterou je možné osadit je 32 GB
  Server byl osazen v základní konfiguraci dostatečným množstvím operační paměti. V

případě potřeby je možné další přidat a to až do velikosti 32GB. Při překročení 3GB RAM je třeba použít 64-bitový operační systém.

- *řadič disků – PERC 5/I – integrovaný*
  Řadič disků patří mezi jedno z nejdůležitějších kritérií při výběru serveru. Je nezbytná jeho podpora pod OS, které chceme provozovat (MS Windows, Linux, FreeBSD atd.). Podstatou vlastností řadiče je jeho výkon, rychlost a spolehlivost.

- *disky – 2× HDD 73GB SAS 15K, 3,5“*
  počet disků je možno rozšířit volitelně až na 6. Z důvodu rychlosti byl vybrán SAS – což je modernější verze SCSI.

- *síťová karta – 2× 1Gbit/s*
  Broadcom NetXtreme II BCM5708
  jedno rozhraní může být použito pro připojení do lokální LAN a druhé do management LAN.

- *grafická karta – integrovaná*

- *CD\DVD – DVD-ROM/CD-RW*
  součástí pro servisní činnost – převážně instalace

6.1.3 Zapojení serveru

Servery KolejNetu jsou umístěny v datových uzlech, které jsou klimatizovány a jsou napájeny přes záložní zdroje napájení.

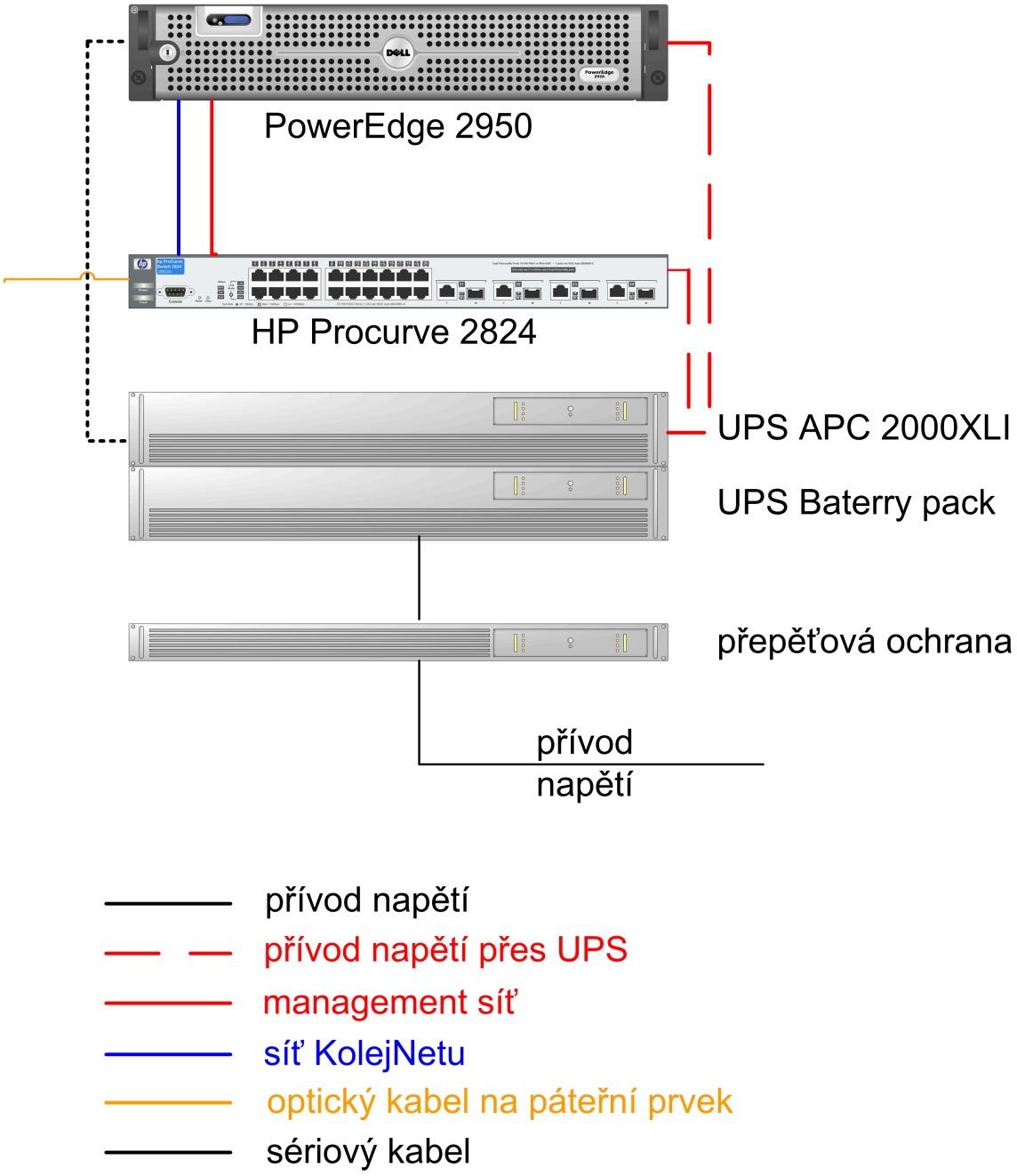
U důležitých serverů je snaha o úplné galvanické oddělení od napájecí sítě. Toto opatření se již několikrát v praxi osvědčilo.

To je realizováno jednak pomocí aktivní UPS a také pomocí síťového prvku, který je ke zbytku infrastruktury připojen pomocí optického propojovacího kabelu.

Tak je zajištěno, že celý rack je galvanicky oddělen a servery jsou ochráněny před poruchami v napájecí i datové části sítě.

Server bude zapojen s těmito komponenty:
- rack DELL 42U
- APC Smart-UPS 2000XLI on-line
- APC Smart-UPS RT 48V BP
- napájecí kabely
- HP Procurve 2824
- HP miniGBIC 1000Base-SX

Konkrétní zapojení nalezneme na obrázku 6.2



Obr. 6.2 Zapojení serveru v uzlu

6.2 Software

Při výběru OS a aplikací je třeba dbát na bezpečnost a znižovat bezpečnostní riziko na minimum, protože umožňuje přístup do management sítě.

6.2.1 Bezpečnostní koncepce

Úroveň zabezpečení počítače, který je zapojen do veřejné počítačové sítě není závislá jen na použitém operačním systému. Podstatným bodem je jeho správná konfigurace a také podmíněna nepřístupnost počítače cizím osobám. Toto bývá velmi často opomíjená bezpečnostní chyba. Operační systém je nejzranitelnější při re/startu).

Čím více překážek, jak programových tak fyzických odděluje útočníka, tím je vyšší šance, že případný útočník neuspěje při útoku.

Proč vlastně více zábran, když by stačila jedna účinná?
Je to dáno lidskými chybami, které se projevují nejen v softwarových produktech. Jejich důsledky lze vidět dennodenně – tisíce oprav u tisíců různých produktů. Opravují se chyby a velmi často svou aplikací zanášejí nové chyby.

Pro zajištění fyzické bezpečnosti budou použity tyto prostředky:
- fyzické zajišťění v datovém uzlu – zabezpečen v uzamčené a střežené místnosti
- uzamknutí serveru v racku
- uzamknutí hlavního panelu serveru
- zabezpečení změny bootování v BIOSu

Pro zajištění bezpečnosti vůči útokům z internetu budou použity tyto ochranné vrstvy:
- firewall sítě KolejNetu
- firewall samotného serveru
- nastavení aplikací běžících na serveru
- přístupová práva k souborům

6.2.2 Operační systém

Z provozovaných OS lze vidět, že převažuje nasazení operačního systému FreeBSD, který se za roky používání osvědčil a správci s ním mají nejvíce zkušeností. Zkušenosti a znalost daného operačního systémem zvyšuje úroveň zabezpečení, při dodržování správné bezpečnostní politiky.
Pro instalaci bude použit operační systém FreeBSD, který je považován za kvalitní systém unixového typu. Má k dispozici potřebné aplikace, široké možnosti zabezpečení a nastavení.

Instalace FreeeBSD
Typická instalace OS je podobná ostatním unixovým operačním systémům . Začíná pořízením OS na médiu, FreeBSD lze zavádět z FDD, sítě či CD-ROM. Proces instalace je popsán v kapitole 1 v [5] , dále se tedy budeme zabývat jen rozdíly v konfiguraci.
Po nabootování k instalaci použijeme program *sysinstall*, který se při standartní instalaci spustí automaticky. Tento program slouží jak k samotné prvotní instalaci, tak možnosti konfigurace či instalace dalších aplikací.

Volba druhu pole
Před samotnou instalací je třeba uvážit, jakým způsobem budeme s disky pracovat. Kolik potřebujeme kapacity, s jakou spolehlivost očekáváme a jak nás zajímá výkon.

V podstatě máme 3 možnosti jak disky použít:
- samostatné použití disků
- softwarový RAID
- hardwarový RAID

Jaké typy RAID existují a jaké mají vlastnosti nalezneme např. zde [18], nebo zde [**?**], kde je daná problematika podrobně rozebrána.

Z praktického hlediska jsou nejpoužívanější tyto druhy:
- RAID 0 – striping – poskytuje rychlost a maximální kapacitu
- RAID 1 – mirroring – jednoduchá a efektivní ochrana dat před ztrátou, za cenu ztráty poloviny kapacity disků
- RAID 5 – striped disks with parity – ochrana dat je zabezpečena distribuovanou paritou, docílíme kapacity n-1 disků.

Před nákupem serveru, byl server zapůjčen a otestován včetně integrovaného RAID řadiče. Výsledky testu jsou uvedeny v podkapitole 6.2.3.

Nastavení RAID

Vybrali jsme vzhledem k rychlosti a spolehlivosti RAID 1 – zrcadlení.
V BIOSu řadiče vybereme příslušný typ pole a necháme jej sestavit. Po sestavení pole ověříme jeho stav a je možné začít instalovat OS.

Rozdělení disků

Systém FreeBSD používá odlišné názvosloví než je obvyklé u ostatních systémů. Místo názvu *partition* se používá *slice*. Ta musí být minimálně jedna, na ní se pak vytvoří oddíl souborového systému. Maximum oddílu na *slice* je 6 a oblast pro *odkládací oddíl*[20].

```
 Part      Mount          Size Newfs
----      -----          ---- -----
mfid0s1a  /             512MB UFS2
mfid0s1b  swap         4062MB SWAP
mfid0s1d  /var         3055MB UFS2+S
mfid0s1e  /tmp         2048MB UFS2+S, nodev, nosuid
mfid0s1f  /usr        16384MB UFS2+S
mfid0s1g  /usr2       43314MB UFS2+S
```

Rozdělení disků včetně velikostí jednotlivých oddílů je vytvořeno podle doporučených hodnot. Oddíl připojený jako */tmp* má navíc atributy *nodev* a *nosuid*, které zajistí, že v adresáři */tmp*, kde mohou zapisovat všichni uživatelé nebude možno vytvářet speciální soubory pro přístup k zařízením a spouštět soubory s právy jiného uživatele. Poslední oddíl je vytvořen zvlášť pro *virtuální stroje* – tzv. JAILy.

Jail je implementace virtualizace pod OS FreeBSD, která umožňuje vytvoření mini-systému na systémové úrovni.

Následně instalační program *sysinstal* nabízí instalaci programového vybavení, zdrojové kódy jádra a balíčkovací systém. Další programy lze buď instalovat nebo zkompilovat dle potřeby.

[20] swap space

Základní konfigurace systému

Update zdrojových kódů jádra

Provedeme ho například pomocí programu *cvsup*, který nainstalujeme třeba pomocí `pkg_add -r cvsup` a upravíme konfigurační soubor, více viz. man cvsup.

Provedeme update stabilní verze a sestavíme podle potřeby nové jádro systému:

1. vytvoříme kopii souboru `/usr/src/sys/i386/conf/GENERIC` se svým názvem např.: `/usr/src/sys/i386/conf/nazev_serveru`

2. nový soubor zeditujeme dle potřeb zvláště přidáme řádky pro podporu IPFW a IPFWv6:

   ```
   #firewall
   options    IPFIREWALL
   options    IPFIREWALL_VERBOSE
   options    IPFIREWALL_VERBOSE_LIMIT=20

   options    IPV6FIREWALL
   options    IPV6FIREWALL_VERBOSE
   options    IPV6FIREWALL_VERBOSE_LIMIT=20
   ```

   Ve starších verzích FreeBSD než 6.× bylo třeba přidat zvláštní parametry, toto se již nastavuje přes `sysctl`.

3. nové jádro sestavíme následovně:
   `cd /usr/src`
   `make buildkernel KERNCONF=nazev`
   možné je přidat parametr `-j4`, pokud máme více procesorů k dispozici.

4. nové jádro nainstalujeme následovně:
   `cd /usr/src`
   `make instalkernel KERNCONF=nazev`

/etc/crontab
Pro chod serveru je nezbytná časová synchronizace serveru se zbylými servery v síti a především s aktivními prvky.

Synchronizaci je možné provést různými způsoby, v tomto případě zvolíme synchronizaci přes cron, což je daemon zajišťující spuštení příkazů ve zvoleném čase.

Do souboru vložíme tento řádek:

```
2 *  *  *  *  root  /usr/sbin/ntpdate ntp.kn.vutbr.cz >/dev/null 2>&1
```

/etc/firewall.conf
Nastavení firewallu je poněkud náročnější a proto se jím budeme zabývat v samostatné sekci.

/etc/skeykeys a /etc/opiekeys
S/Key je autentizační mechanismus povolujici OPIE více v manuálových stránkách `skey`.

```
touch /etc/skeykeys
touch /etc/opiekeys
/bin/chmod 600 /etc/skeykeys
/bin/chmod 600 /etc/opiekeys
```

/etc/profile
Soubor zajišťuje nastavení proměnných uživatelského prostředí:
`export PS1="[\\u@\\h \\w]\\\$"`

/etc/rc.conf
Tento soubor je základní konfigurační soubor pro nastavení systému a v našem případě bude jeho obsah vypadat takto:

adresa defaultního směrovače – nastavení směrovací tabulky
`defaultrouter="147.229.190.129"`

jméno počítače
`hostname="sea.kn.vutbr.cz"`

nastavení IP adresy a síťové masky
`ifconfig_bce0="inet 147.229.190.190 netmask 255.255.255.128"`

nastavení IP adresy a síťové masky
`ifconfig_bce1="inet 172.23.0.190 netmask 255.255.255.128"`

zapnutí sledování procesů
`accounting_enable="YES"`
zapnutí firewallu
`firewall_enable="YES"`

konfigurace firewallu
`firewall_type="/etc/firewall.conf"`

zapnutí logování firewallu
`firewall_logging="YES"`

vypnutí inetd
`inetd_enable="NO"`

vypnutí emulace linuxu – využijeme až v JAILU
`linux_enable="NO"`

zapnutí ssh daemona
`sshd_enable="YES"`

zapnutí sedmailu v klientském režimu
`sendmail_enable="NO"`

zapnutí USB – klávesnice
`usbd_enable="YES"`

/etc/resolv.conf
V tomto souboru nastavíme doménu a adresy DNS serverů přidáním těchto řádků:
```
domain kn.vutbr.cz
nameserver 147.229.191.135
nameserver 147.229.190.134
nameserver 147.229.3.10
```

/etc/sysctl.conf
Soubor `sysctl.conf` se používá ve víceuživatelském módu pro základní nastavení jádra. Podrobnosti zjistíme v `sysctl -a -d` a v manových stránkách.

Je třeba zkontrolovat toto nastavení:
```
kern.maxfiles
kern.maxproc

security.bsd.see_other_uids=0
net.inet.tcp.log_in_vain=1
net.inet.udp.log_in_vain=1
net.inet.tcp.blackhole=1
net.inet.udp.blackhole=1
```

Tyto volby zajistí následující:

1. zamezí vidět uživatelským účtům informace o procesech, který běží pod jinými UID
2. zapnutí protokolování všech pokusů o připojení na TCP porty, na kterých nenaslouchá žádná aplikace
3. zapnutí protokolování všech pokusů o připojení na UDP porty, na kterých nenaslouchá žádná aplikace
4. zahodí pokusy k TCP uzavřeným portů, bez odpovědi *connection reset by peer*
5. zahodí pokusy k UDP uzavřeným portů, bez odpovědi

Nastavení firewallu

Bezpečnost serveru bude zajištěna dvěmi firewally, první bude samotný firewall serveru, který bude zajišovat primární bezpečnost a druhý bude firewall sítě Kolejnetu, který bude zajišťovat ochranu serveru před útoky z internetu.

Server má 2 síťové karty, jednou bude připojen do sítě Kolejnet a světa, a druhou bude připojen do management sítě. Tato síť je zcela oddělena od Internetu a je třeba důkladně zajistit bezpečnost právě tohoto serveru.

Firewallu serveru

K dispozici jsou tyto firewally: PF[21], IPFW [22]. Tenty firewally jsou součástí jádra OS serveru. V tomto případě byl zvolen jako firewall IPFW.

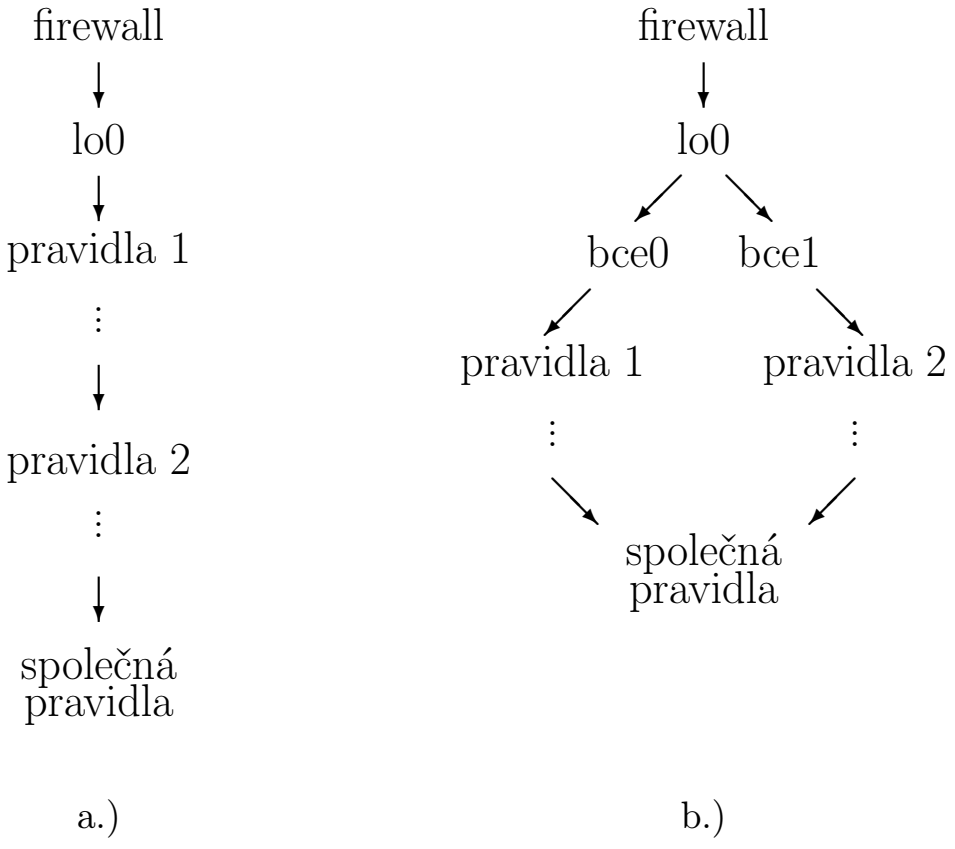
Tímto firewallem projde každý paket z obou síťových karet. Firewall je koncipován jako uzavřený a cílem je, aby server mohl komunikovat se světem z důvodu aktualizací programového vybavení serveru.

Současně bude nastavena bezpečnostní politika silně restriktivně vůči komunikaci z internetu až na vyjimky, které budou podle potřeby definovány.

Stejně tomu bude v management síti, která je sice bezpečná, ale není důvod snižovat úroveň zabezpečení vůči ní.

Protože budeme přijímat velké množství paketů hlavně na rozhraní management sítě, rozdělíme pravidla podle rozhraní poté co projdou rozhraním lo0[23].

Zásadně tak omezíme počet pravidel, která se budou procházet → menší zátěž pro server.



Obr. 6.3 Průchod paketů firewallem: a.) standardní nastavení firewallu b.) optimalizovaná verze firewallu

[21] známý jako OpenBSD's PacketFilter
[22] známý jako: IPFILTER, IPFIREWALL
[23] tzv. loopback

Současně nám umožní přesněji definovat pravidla a dojde současně k jejich zpřehlednění. Rozdělení je patrné z obrázku 6.3.

Firewall bude také zaznamenávat neúspěšné pokusy a následně o nich budeme informováni v logovacím souboru.

Obvykle se používá v pravidlech pro vnější rozhraníme namísto IP adresy, ale tento server má více než jedno rozhraní. Proto bude použito pravidlo, které je jednoznačné – IP adresa.

Pravidla firewallu jsou uvedeny v souboru */etc/firewal.conf*.

Začátek společných pravidel:

```
add 100 allow ip from any to any via lo0
```

```
add 200 deny ip from any to 127.0.0.0/8
```

```
add 300 deny ip from 127.0.0.0/8 to any
```

Předchozí pravidla zajistí, aby aplikace mohly spolu komunikovat pomocí vnitřního rozhraní lo0 (síť 127.0.0.0/80), ale současně zajistí, aby se nebylo možné dostat do této sítě přes jiná rozhraní a stejně z této sítě ven.

```
add 500 skipto 10000 all from any to any via bce1
```

```
add 600 skipto 1000 all from any to any via bce0
```

Odskok pravidel podle rozhraní 2 – bce1 a 1 – bce0.

Začátek pravidel pro rozhraní 1 – bge0:

```
add 1000 allow udp from 147.229.190.134 53 to 147.229.190.190 1024-65535
```

```
add 1300 allow udp from 147.229.191.135 53 to 147.229.190.190 1024-65535
```

Povolení komunikace DNS s primárním a sekundárním serverem.

```
add 1400 allow udp from 147.229.191.135 123 to 147.229.190.190 123
```

Umožnění komunikace s NTP serverem.

```
add 2000 allow tcp from any to me established
```

Pravidlo povoluje již navázaná spojení přes TCP.

```
add 2100 allow tcp from me to any
```

Povolení odchozího TCP provozu.

```
add 2200 allow tcp from 147.229.220.x to 147.229.190.190 22

add 2300 allow tcp from 147.229.220.2 to 147.229.190.190 22

deny 2400 deny tcp from any to 147.229.190.190 22
```
Povolení přístupu z námi zvolených přes ssh a zakázání zbylých.

```
add 3000 allow ip from 147.229.190.190 to any
```
Povolení pro komunikaci směrem ven.

```
add 9999 skipto 20000  all from any to any via bce0
```
Odskok na společná pravidla.

Začátek pravidel pro rozhraní 2 – bge1:

```
add 10100 allow udp from 172.16.0.0/13 to 172.23.0.190

add 10200 allow udp from 172.23.0.190 to 172.16.0.0/13
```
Pravidla pro UDP provoz, zde bude největší zátěž tvořená SNMP.

```
add  11000 allow tcp from any to 172.23.0.190 established
```
Povolení již navázaného TCP provozu.

```
deny 11200 deny tcp from any to me 22
```
Zákaz přístupu přes ssh.

```
add  11300 allow ip from 172.23.0.190 to any
```
Povolení odchozího provozu přes rozhraní 2.

```
add 19999 skipto 20000  all from any to any via bce1
```
Odskok na společná pravidla.

Začátek společných pravidel pro obě rozhraní:

```
add 20100 deny log logamount 0 tcp from any to me 1-1024
```
Zahození všech příchozích paketů v daném rozsahu, které jsou přidělovány spouštěným službám a logují se pokusy přístupu k nim.

```
add 20200 deny log logamount 0 ip from any to me
```
Zahození a logování všech ostatních paketů na protokolu IP.

Pravidla lze samozřejmě podle potřeby přidávat vždy s ohledem na pořadí. Jinak by mohlo dojít k degradaci výkonu serveru při průchodu paketů firewallem.

Firewall KolejNetu

Tento firewall zajišťuje oddělení sítě KolejNet od sítě VUT a Internetu. Je realizován pomocí PC (OS CentOS) a HP Procurve 6108.

Mezi pravidla bude přidáno pravidlo: zakáz přístupu na IP adresu vnějšího rozhraní a port 22, 5432 (ssh, postgres).

Nastavení sshd

Jedná se o daemona ssh, který umožňuje vzdálené přihlášení k systému. Daemon sshd je součástí standardní instalace FreeBSD, proto ho není třeba instalovat.

O start se postará skript `/etc/rc.d/`, který je spuštěn na základě nastavení v souboru `/etc/rc.conf/`.

Samotná konfigurace sshd je v souboru `/etc/ssh/sshd_config/`, zde uvedeme jen rozdíly od výchozí konfigurace, které jsou nezbytné pro bezpečné provozování vzdáleneho přístupu přes protokol ssh.

Konfigurace:

`ListenAddress 147.229.190.190`

Určíme konkrétní adresu na které se bude daemno naslouchat.

`PermitRootLogin yes`

Povolíme přihlášení uživatele root, jinak je nutné vytvořit uživatele a přihlašovat se přes příkaz `su`. Toto budeme kompenzovat nastavením firewallu a přihlášením jen pomocí klíče.

`PermitRootLogin without-password`

Zakáže přihlášení roota heslem.

`ChallengeResponseAuthentication no`

Touto volbou je zakázána PAM autentifikace.

`PermitEmptyPasswords no`

Zakázání přihlašení prázdným heslem.

`StrictModes yes`

Zajistí kontrolu domovského adresáře:
*/rhost /.shost /.ssh /.ssh/authorized_keys*2 */.ssh/authorized_keys*2

Zjištění fingerprintu[24] *sshd*

DSA klíč na serveru. Pokud ho neznáme vygenerujeme ho, návod je v další sekci.

```
ssh-keygen -l -f /etc/ssh/ssh_host_dsa_key.pub
2048 22:ff:f8:22:...:82:b2:d3:28:15 /etc/ssh/ssh_host_dsa_key.pub
```

[24] "otisk" klíče, obvykle pomocí MD5 a SHA1

Práce s klíči

1. Generování klíčů pro sshd, spustíme následující skript:
   `/etc/rc.d/sshd sshd_keygen`
   vygeneruje se veřejný a soukromý klíč

2. zkontrolujeme fingerprinty – musí souhlasit, poté se přihlásíme heslem.

3. Na svém desktopu vygenerujeme veřejný klič.pub a soukromý klíč:
   `ssh-keygen -b 4096 -t dsa -f jméno_klíče`

4. Přidáme soukromý klíč do ssh agenta
   `ssh-add jméno_klíče`

5. Překopírujeme veřejný klíč na server
   `scp jméno_klíče.pub root@server:~/.ssh/authorized_keys`
   a nastavíme práva
   `~/.ssh/authorized_keys jen pro čtení`

6. Vyzkoušíme přihlášení na server
   `ssh root@server`
   Reload sshd vzdáleně `nohup killall -HUP sshd &`

Nastavení upsd

Upsd je daemon, který zajišťuje pomocí ovladače komunikaci mezi ovladačem a klienty. Ovladače jsou programy, které zajišťují vlastní komunikaci s UPS.

Klient je aplikace, která prostřednictvím upsd monitoruje stav UPS a na základě stavu vykonává příslušné akce. Příkladem toho je shutdown při výpadku napájení.

1. instalace deamona
   `/usr/ports/sysutils/apcupsd`
   `make && make install`

2. konfigurace se provede v souboru `/usr/local/etc/apcupsd/apcupsd.conf`

   `UPSTYPE apcsmart`
   : typ UPS

   `UPSCABLE smart`
   : typ kabelu, kterým je UPS připojena

   `DEVICE /dev/cuad0`
   : nastavení portu na kterém je UPS připojena

   `SCRIPTDIR /usr/local/etc/apcupsd`
   : cesta k adresáři, kde jsou umístěny skripty

`PWRFAILDIR /var/run`
cesta k adresáři, kde se vytvoří příznak výpadku napájení

`ONBATTERYDELAY 6`
čas v sekundách, který nebude považován za výpadek

`BATTERYLEVEL 5`
čas v minutách zbývající do vyčerpání baterií, poté bude spuštěna ukončovací sekvence

`MINUTES 3`
cesta k adresáři, kde jsou umístěny skripty

`NETSERVER off`
zakázání spouštění síťového serveru

`UPSNAME UPS-PPV-2`
jméno UPS – použije se při zápisu do logů a zprávách zasílaných e-mailem

`SENSITIVITY M`
citlivost na hodnoty napětí, pro přepnutí na baterie

`SLEEP 120`
prodleva před odpojením spotřebičů po zadání příkazu

`RETURNCHARGE 15`
nabití baterií v procentech, před obnovením napájení

`OUTPUTVOLTS 230`
nastavení požadovaného výstupního napětí

6.2.3 Testování rychlosti

Testování výkonu serveru a hledání limitů je důležitá činnost a v této sekci si uvedeme důvody proč je vhodné je provádět.

Testy je vhodné provést před samotným nákupem serveru, nebo bezprostředně po nákupu, nemáme-li reference o výkonu z nějakého nezávislého zdroje.

Co nám testování nabízí a zajišťuje:

návrh monitorovacího systému
Je třeba navrhnout a provozovat aplikace, které při běžné zátěži je schopen server obsloužit.

bezpečný provoz
Pro provoz potřebujeme mít jistou míru přebytku výkonu, aby se server byl schopen vyrovnat s běžnými provozními problémy. Při běžném provozu nastávají různé kritické

situace od výpadku napájejí až různé útoky. Server by měl tyto situace zvládnout a to bez výpadku služeb, nebo co s nejmenším (nejkratším) výpadkem.

rozšíření počtu prvků
Je nezbytné počítat s rozšíření počtu sledovaných prvků a proto musí být splněna alespoň jedna z těchto podmínek:

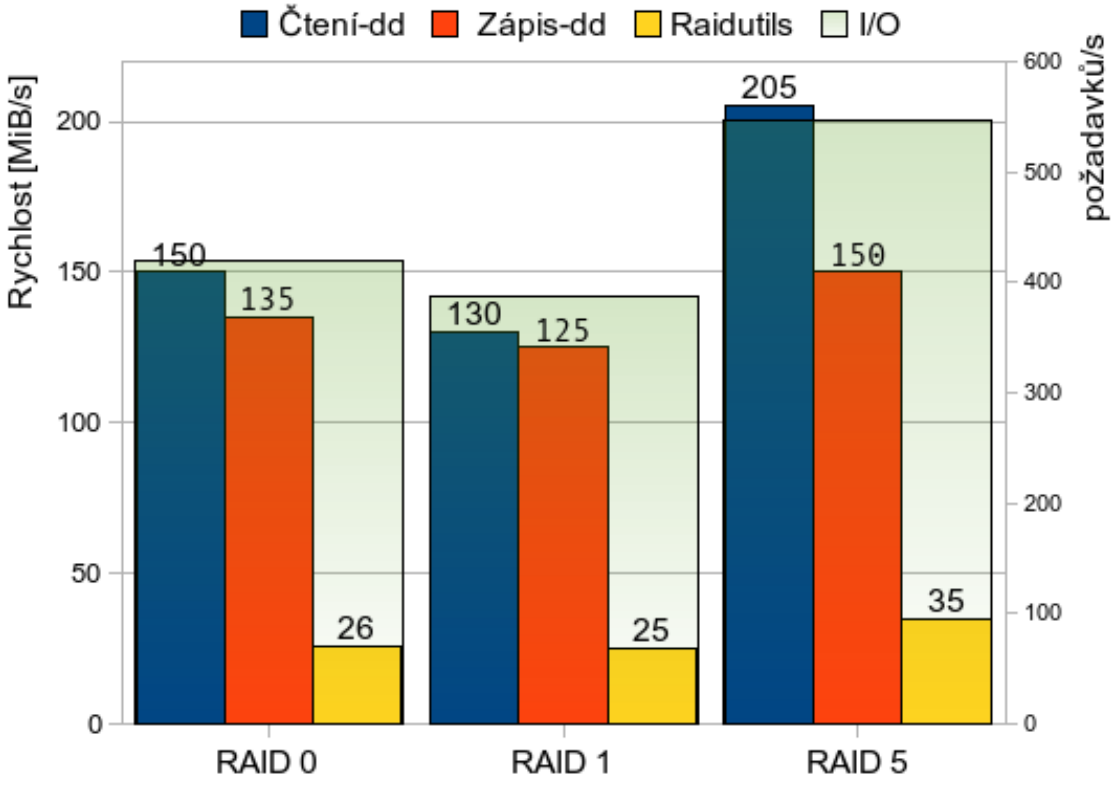
- server musí být na tento nárůst dimenzován
- monitorovací systém musí být navržen tak, aby se dokázal s tímto nárůstem vyrovnat
- musí být znám odhad, kolik je server schopen bezpečně obsloužit prvků

výměny hardwaru serveru
Musíme si být jisti, že při změně hardwaru bude monitorovací systém fungovat a bude mít k dispozici dostatečný výkon.

6.2.4 Testování kritických oblastí

- výkon CPU
- propustnost pamětí
- výkon diskového subsystému
- výkon NIC

Testování rychlosti diskového subsystému bylo provedeno dvěmi testovacími nástroji dd a *raidtest*.



Obr. 6.4 Testy řadiče PERC5i v závislosti na typu RAID

dd

Příkaz dd je součástí OS a jeden proces provádí lineární čtení/zápis.

Test proveden příkazem pro čtení:
`dd if=/dev/mfid0 of=/dev/null bs=1m`

a pro zápis:
`dd if=/dev/random of=/dev/mfid0 bs=1m`.

raidtest

Pro test byla použita utilita *raidtest*. Test spočívá v současném běhu 10 procesů, které provádí paralelní čtení/zápis.

```
set mediasize=‘diskinfo /dev/mfid0 | awk ’{print 3}’‘
set sectorsize=‘diskinfo /dev/mfid0 | awk ’{print 2}’
raidtest genfile -s mediasize -S sectorsize -n 50000
raidtest test -d /dev/mfid0 -n 10
```

```
RAID:   R   W
0: 120 120
1: 130 130
5: 205 150
```

```
set mediasize=‘diskinfo /dev/mfid0 | awk ’{print 3}’‘
set sectorsize=‘diskinfo /dev/mfid0 | awk ’{print 2}’
raidtest genfile -s mediasize -S sectorsize -n 50000
raidtest test -d /dev/mfid0 -n 10
```

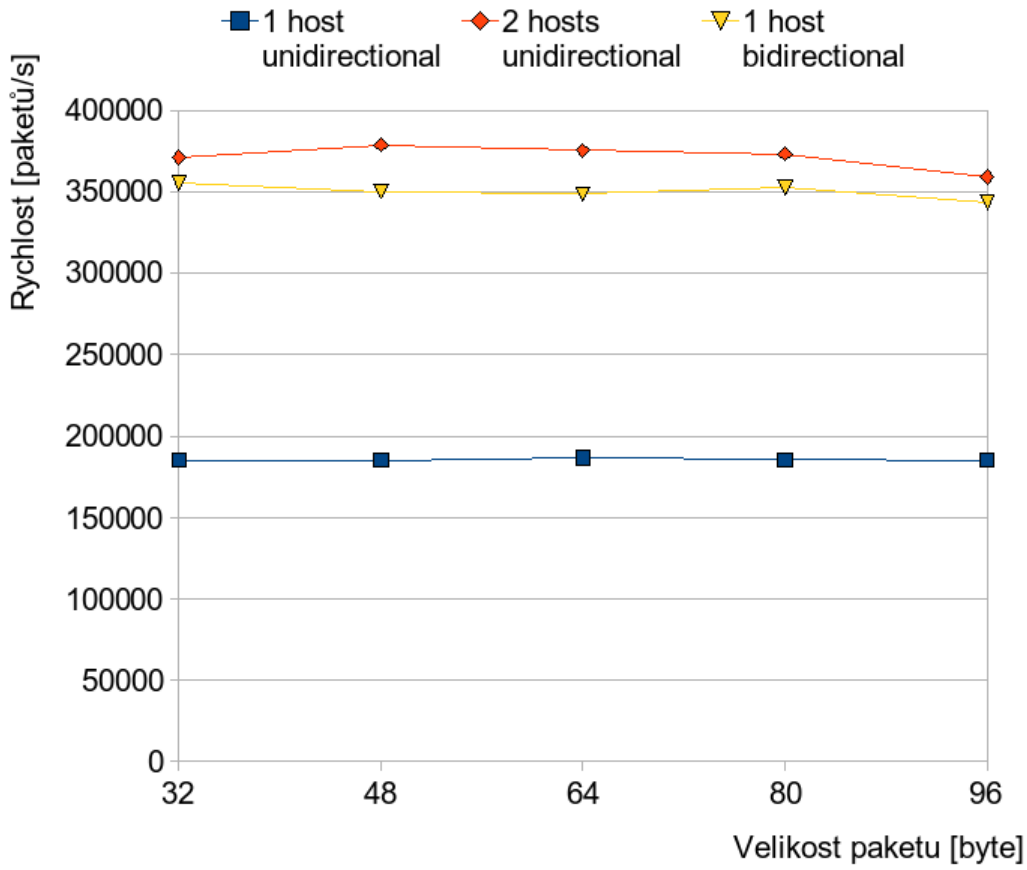
```
sea# raidtest test -d /dev/mfid0 -n 10
Read 50000 requests from raidtest.data.
Number of READ requests: 24947.
Number of WRITE requests: 25053.
Number of bytes to transmit: 3300104704.
Number of processes: 10.
Bytes per second: 25011025
Requests per second: 378
```

```
baja# raidtest test -d /dev/mfid0 -n 10
Read 50000 requests from raidtest.data.
Number of READ requests: 24847.
Number of WRITE requests: 25153.
Number of bytes to transmit: 3291225088.
Number of processes: 10.

Bytes per second: 35993326
Requests per second: 546
```

Testování propustnosti UDP

Aby mohl být v reálném provozu používán protokol SNMP verze 2, je třeba zjistit chování při průchodu velkého množství UDP paketů.



Obr. 6.5 Průchod UDP paketů firewallem

Existuje řada testovacích nástrojů – k testu použijeme prověřený nástroj *netperf*.

Budeme testovat UDP pakety o různých délkách paketů a to podle obvyklé délky, kterou používá SNMP.

Na serveru spustíme `netserver -4 -L 147.229.190.190 -p 22113` a na klientovi `netperf` s patřičnými parametry.

6.3 Návrh sledovacího software

Návrh musí respektovat stávající situaci v síti KolejNet. Z toho vyplývá, že není možné měnit např. topologii sítě. Při návrhu je nutné dbát na bezpečnost, protože se jedná o přístupový bod do management sítě. Z něj lze narušit celou bezpečnostní koncepci sítě.

Vycházíme z faktu, že máme k dispozici 2 sítě (běžný provoz a management), které jsou souběžné a je vhodné využít vlastností, které nabízejí.

Shromážděná data by musí být dostupná pro IS KolejNetu v reálném čase, aby mohl být použit pro vyhodnocení nastalých situací vzhledem k uživatelům.

Snahou bude, aby útoky, které jsme uvedli byly detekovány a mohly být zpětně dohledatelné. To samé může nastat při nějakých nezvyklých anomáliích.

Shromážděná data mohou v budoucnu poskytnout cenné informace, pro odhalení chyb. Pokud je možné budeme se zabývat i eliminací možných útoků. Pokud můžeme něčemu zabránit, můžeme to vypustit ze sledování.

Pro správu prvků je vhodná oddělená síť pro management hlavně z důvodů nezávislosti na běžném provozu. Bude tedy fungovat i během případného útoku, který se v tak velké síťi obtížně lokalizuje. Lokalizace takového útoku je obtížná a vyžádá si svůj čas. Útoky nemusí vždy trvat kontinuálně, ale mohou být probíhát ve vlnách, které nemusí být periodické.

Než jsou vypátrany zdroje útoku, může útok ustat a viník nebo příčina zůstane skrytá.

Budeme mít požadavky na sledovací software, který nám umožní lokalizaci zdrojů útoků i po odeznění samotného útoku.
Mezi další důležité vlastnosti je monitoring vlastníků IP adres ve vztahu k MAC adresám, které jsou v síti přiřazovány DHCP servery podle dat v IS KolejNetu.

IP a MAC adresy ovšem nestačí pro pro identifikaci. MAC adresa má být unikátní podle IEEE[25] a jejich standardů, ale běžně se vyskytují duplicity. Také není složité změnit MAC adresu za provozu přímo na síťové kartě.

Potřebujeme tedy informaci o tom, jaká MAC adresa byla užívaná v určité chvíli na určitém prvku a portu. Dále nás bude zajímat jestli byl port aktivní a bylo k němu připojeno nějaké zařízení.

Útoky se obvykle vyznačují atypickým chování, obvykle je snaha o zaplavování sítě různě modifikovanými pakety. Bude nás tedy zajímat odchozí traffic, počet paketů a vytížení aktivních prvků.

Vzhledem k počtu provozovaných prvků je vhodné archivovat jejich konfiguraci a udržovat jejich aktuální stav. Také je vhodné logovat podezřelé události na prvcích.

6.3.1 Souhrn požadavků bodech

Musí být dostupné následující informace:
1. zpětná dostupnost událostí
2. použití MAC-IP v čase
3. MAC lokalizovatelná v síti vzhledem k portu určitého prvku
4. aktivita na jednotlivých portech
5. vytížení aktivních prvků
6. logování podezřelých aktivit

Shromažďované informace budou získávány z aktivních prvků pomocí protokolu SNMP verze 2c a to za podmínky, že komunikace bude probíhat po management síti.

[25] The Institute of Electrical and Electronics Engineers – mezinárodní organizace standardů elektroinformatiky, viz http://www.ieee.org

6.3.2 Realizace požadavků

Zpětná dostupnost událostí
Pro sběr a zpracování dat se jeví jako nejvhodnější řešení databázový systém. Zjednodušuje běžné operace s daty snadno se udržuje, včetně záloh a poskytuje rychlost a komfort při práci s daty.

Jako programovací jazyk, který umožňuje snadné spuštění externích programů a snadnou komunikaci DBS je Perl.

Sledování použití IP a MAC adres
Tyto informace budou shromažďovány z jednotlivých routerů, což jsou centrální prvky v každém areálu.

Lokalizace MAC adres na portech
Data budou čerpány periodicky z jednotlivých aktivních prvků.

Aktivita na portech
Z aktivních prvků budou shromažďovány informace o zátěži, která je na nich generována.

Logování podezřelých aktivit
Z každého prvku bude shromažďovány informace o podezřelých aktivitách.

Logování podezřelých aktivit
Podezřelé aktvivity budou logovány a to pomocí trapů, které aktivní prvky podporují.

6.3.3 Výběr a instalace DBS

Prvně než dojde k instalaci, je třeba zvážit, kterou DBS použít. K výběru máme několik možností, které jsou uvedeny v tabulce 6.1.

Tab. 6.1 Vlastnosti DBS

DBS	Verze	Licence	SQL92	SQL99	SQL2003	Perl
PostgreSQL	8.3	BSD	x	x	x	x
MySQL	5.0	GPLv2,3	x	x	x	x
SQLite	3.6	public domain	x			x
Firebird	2.1	IDPL	x			x

DBS uvedené v tabulce 6.1 jsou si svým výkonem a možnostmi velmi blízké (až SQLite). Z výše uvedených jsou na KolejNetu používány, jak je uvedeno v tabulce 6.2 MySql a PostgreSQL.

Pro zjednodušení správy je vhodné volit právě mezi těmito dvěma, které se již používají. Obě jsou pro tento účel použitelné, ale PostgreSQL má řadu funkcí delší dobu a do MySql byly některé z nich doplněny teprve nedávno.

Pro IS je použito PostgreSQL a protože tento dohled se v budoucnu stane součástí IS, volíme pro použití PostgreSQL.

Instalace PostgreSQL

Instalaci provedeme z portů v defaultní konfiguraci.

```
cd /usr/ports/databases/postgresql82-server/
make WITH_OPTIMIZED_CFLAGS=yes install
```

Aby PostgresSQL mohlo používat interface, je třeba do konfiguračního souboru zadat `listen_addresses = '$ip'`. Za `$ip` bude zadána IP adresa ze kterém chceme přístup do databáze.

Přístup do databáze, pokud PostgresSQL bude naslouchat na vnějším rozhraní, je vhodné omezit na firewallu jen na IP adresy, ze kterých chceme mít přístup a ostatní zakázat.

To provedeme vložením těchto pravidel na vhodnou pozici zadáním čísla s ohledem na pravidla, které se používají.

Změnu provedeme jak z příkazového řádku a po ověření editací `/etc/ipfw.conf`:

```
add číslo allow tcp from ip1 to me 5423
add číslo allow tcp from ip1 to me 5423
add číslo deny tcp from any to me 5423
```

Za *ip1* a *ip2* zadáme IP adresy, ze které chceme přistupovat k databázi.

Konkrétní IP adresu rozhraní, na kterém naslouchá vložíme za *me*. Tím zajistíme, že se bude pravidlo aplikovat jen na rozhraní na kterém je pravidlo požadováno.

Před inicializací databáze je třeba vložit do `/etc/login.conf` nasledující:

```
postgres:\
        :lang=en_US.UTF-8:\
        :setenv=LC_COLLATE=C:\
        :tc=default:
```

Nastavení zajistí porovnávání řetězců podle zvolené znakové sady a nastavení jazyku podle potřeby. V našem případě nepotřebujeme měnit vlastnosti předurčeného nastavení.

`LC_COLLATE`
: porovnávání řetězců

`LC_CTYPE`
: klasifikace znaků

`LC_MESSAGES`
: překlady zpráv

`LC_MONETARY`
: formátování zápisu u peněz

`LC_NUMERIC`
: formátování zápisu u čísel

`LC_TIME`
: formátování zápisu času/datumu

Proto, aby se nastavení projevilo je třeba spustit následující příkaz:
```
cap_mkdb /etc/login.conf
```

Pro spouštění databáze po startu je třeba editovat `/etc/rc.conf` a vložit tento řádek:
```
postgresql_class="postgres"
postgresql_enable="YES"
```

Před spuštěním je třeba pro daný operační systém nastavit sdílenou paměť a semafory[26]. V případě FreeBSD je třeba editovat */etc/sysctl.conf* a */boot/loader.conf* a poté provést restart systému.

Editace */etc/sysctl.conf*:
```
sysctl -w kern.ipc.shmall=32768
sysctl -w kern.ipc.shmmax=134217728
sysctl -w kern.ipc.semmap=256
sysctl -w kern.ipc.shm_use_phys=1
```

Pokud akce zdaří je možně vidět toto:
```
# sysctl -w kern.ipc.shmall=32768
kern.ipc.shmall: 8192 -> 32768
# sysctl -w kern.ipc.shmmax=134217728
kern.ipc.shmmax: 33554432 -> 134217728
# sysctl -w kern.ipc.semmap=256
kern.ipc.semmap: 30 -> 256
```

Editace */boot/loader.conf*:
```
set kern.ipc.semmni=256
set kern.ipc.semmns=512
set kern.ipc.semmnu=256
```

Inicializujeme databázi:
```
/usr/local/etc/rc.d/postgresql initdb
```

Zajistíme spuštění databáze po startu operačního systému vložením tohoto řádku do `/etc/rc.conf`:
```
postgresql_class="postgres"
```

Databázi spustíme:

```
/usr/local/etc/rc.d/postgresql start
```

Vytvoření a konfigurace databáze

Vytvoříme databázi *ppv* a uživatele *scan*, který bude s databází pracovat.

```
CREATE DATABASE  ppv;
CREATE USER scan;
```

[26] System V IPC

```
ALTER USER scan WITH PASSWORD 'heslo';

CREATE GROUP scang WITH USER scan;
GRANT ALL privileges on database ppv to scan;
\connect ppv scan
```

Databáze je navržena obecně a připouští v případě potřeby změny. Slouží pro zachycení jednotlivých událostí a možnosti vyhledávání v nich dle potřeby. Databáze obsahuje celkem 10 tabulek, které nejsou z výkonnostních důvodů vzájmeně spjaty.

Tabulky a sloupce mají tento význam:

model

Zde jsou uvedeny údaje o typu, modelu prvků a jejich maximálním počtu portů.

- *model* je model prvku daný výrobcem
- *type* je typ prvku daný výrobcem
- *num_ports* maximální počet portů daného modelu

pass

Je zde uvedeno community name, což je obdoba hesla a jeho identifikační číslo.

- *id* je identifikátor cname
- *cname*je community name

vlan

V této tabulce jsou uvedeny informace o VLAN, id VLAN a interním id VLAN prvku, který prvek používá interně. Dále je zde uveden příznak, zda se má určitá VLAN scanovat a probíhat sběr dat.

- *id* je identifikátor VLAN
- *name* je název VLAN
- *scan* je příznak, zda se z určité VLAN provádět sběr dat
- *ifid* je interní id VLAN prvku

switch

Tato tabulka je největší a obsahuje podstatnou část informací o aktivních prvcích v síti.

- *ip* je IP adresa prvku
- *location* je pozice umístění prvku
- *sn* je seriové číslo prvku
- *swrev* je softwarová revize – firmware
- *romver* je verze ROM
- *ntp* IP je adresa NTP serveru, kterou prvek používá
- *scan* je příznak sběru dat z prvku
- *sysname* je systémové jméno

- *uplink* je číslo portu uplinku
- *mng* je číslo portu management sítě

usage
V tabulce jsou uvedeny prvky pokud jejich vytížení přesáhlo zvolenou hranici.

- *ip* je IP adresa prvku
- *ts* je čas, kdy došlo vytížení prvku
- *cpu* je procentuální vytížení

ip_mac
Obsahuje seznam aktivních IP adres a záznamů, které jsou v FDB tabulce.

- *ip* je IP adresa
- *mac* je MAC adresa
- *ts_start* je čas ve kterém byl nalezen první záznam
- *ts_end* je aktuální čas
- *del* je příznak vymazání

im_arch
Obsahuje seznam neaktivních IP adres a záznamů, které byly přesunuty po vypršení záznamu z tabuky *ip_mac*.

- *ip* je IP adresa
- *mac* je MAC adresa
- *ts_start* je čas ve kterém byl nalezen první záznam
- *ts_end* je čas ve kterém byl nalezen poslední záznam

mac_port
Obsahuje seznam aktivních MAC adres a doprovodné informace.

- *ipsw* je IP adresa prvku na které byla MAC adresa nalezena
- *port* je port prvku na které byla MAC adresa nalezena
- *ts_start* je čas ve kterém byl nalezen první záznam
- *ts_end* je aktuální čas
- *del* je příznak vymazaní

mac_port_arch
Obsahuje seznam neaktivních MAC adres a doprovodné informace.

- *ipsw* je IP adresa prvku na které byla MAC adresa nalezena
- *port* je port prvku na které byla MAC adresa nalezena
- *ts_start* je čas ve kterém byl nalezen první záznam
- *ts_end* je čas ve kterém byl nalezen poslední záznam

ports
Tabulka obsahuje informace o aktivitě portů.

- *ip* je IP adresa prvku
- *port* je číslo portu prvku
- *ts* je čas pořízení záznamu
- *online* je příznak aktivity portu
- *lock* je příznak uzamčeného portu
- *byte_rx* je čítač přijatých dat prvkem
- *byte_tx* je čítač odeslaných dat prvkem
- *uni_rx* je čítač přijatých dat – typ unicast
- *bm_rx* je čítač přijatých dat – typ broadcast/multicastových

```
CREATE TABLE model (
        model varchar 16,
        type varchar 20,
        num_ports int2,
);

CREATE TABLE pass (
        id int2,
        cname varchar 16
);

CREATE TABLE vlan (
        id_vlan INT,
        name_vlan CHAR(8),
        scan_vlan BOOLEAN
);

CREATE TABLE switch (
        ip inet PRIMARY KEY,
        location varchar(32),
        sn varchar(16) UNIQUE,
        swrev varchar(16),
        romver varchar(16),
        ntp inet,
        model varchar(16),
        scan bool,
        sysname varchar(32) NOT NULL,
        uplink int2
        mng int2
);



CREATE TABLE usage (
        ip inet,
        ts timestamp,
        cpu int2
);
```

```
CREATE TABLE ip_mac (
        ip inet,
        mac macaddr,
        ts_start timestamp,
        ts_end timestamp,
        del bool
);

CREATE TABLE im_arch (
        ip inet,
        mac macaddr,
        ts_start timestamp,
        ts_end timestamp,
        del bool
);



CREATE TABLE mac_port (
        ipsw inet,
        port int2,
        mac macaddr,
        ts_start timestamp,
        ts_end timestamp,
        del bool
);

CREATE TABLE mac_port_arch (
        ipsw inet,
        port int2,
        mac macaddr,
        ts_start timestamp,
        ts_end timestamp
);

CREATE TABLE ports (
        ip inet,
        port int2,
        ts timestamp,
        online bool,
        lock bool,
        byte_rx bigint,
        byte_tx bigint,
        uni_rx bigint,
        bmcast bigint,
);
```

Program

Programy jsou napsány v jazyku Perl, který je standardně součástí FreeBSD a není jej nutné doinstalovávat.

Pro komunikaci s aktivními prvky byl použit protokol SNMP verze 2c. Oblíbenou utilitou pro komunikaci pomocí protokolu SNMP je balíček *net-snmp*. Tato utilita byla použita pro ověření komunikace s aktivními prvky, ověření voleb OID a získaných hodnot.

Pro komunikací programu s databází bylo použito modulu DBI, který je součástí jazyku perl.[27].

Původně bylo počítano s využitím utility *net-snmp* pro "dolováním dat', to se však ukázalo jako nevhodné řešení. Protože v případě stovek prvků a získávání desítek hodnot, by se jednalo o několik tisíc procesů.

Proto byl použit modul Net::SNMP, který umožnil získávání libovolného počtu hodnot jedním procesem pro každou aplikaci.

Zmenšila se tak nejen paměťová a výkonnová náročnost, ale zmenšil se i počet procesů. Zvýšila se tak přehlednost běžících procesů a úloh. Schéma sběru dat a nastavení prvku programy je na obrázku 6.6

Obr. 6.6 Schéma fungování programů

Jednotlivé programy se rozdělují na hlavní a pomocné. Návod k použití každého z nich je uveden v nápovědě a lze ji vyvolat před jeho spuštěním zadáním parametru *-h*. Jména hlavních programů odpovídají tabulkám, které se primárně obsluhují.

Hlavní programy získavají data nepřetržitě z aktivních prvků a získaná data se ukládají do databáze.

Programy pracují nepřetržitě a řídí se systémovým časem. Běží ve smyččce a aktivují akce podle nastaveného času. Po ukončení cyklu vyčkávají do doby, než má proběhnout opakování příslušné akce. Tím je zajištěno přesné a periodické zpracování dat z prvků a ošetřena možnost před zahlcením aktivních prvků požadavky.

Obr. 6.7 Princip zachytávání alertů

Pomocné programy slouží převážně pro vkládání, mazání a zobrazování dat.

[27] rozhraní pro přístup k databázím

Seznam programů:

`mac-ip.pl`
Běží nepřetržitě a zjišťuje dvojici MAC-IP z vybraných VLAN routeru a ukládá je do tabulky mac-ip. Pokud není záznam platný, přesune jej do tabulky *im_arch*.

`mac-port.pl`
Funguje nepřetržitě a zjišťuje MAC adresy na kterých portech prvků jsou použity, ukládá je do tabulky mac-port. Neplatný záznam přesune do tabulky *mac_port_arch* .

`ports.pl`
Běží nepřetržitě a plní tabulku *ports* údaji.

`vlan.pl`
Zajišťuje operace s taboulkou vlan: zobrazení, vkládaní a mazaní v tabulce *vlan*.

`switch-link.pl`
Zajišťuje manipulace s čísly portů uplinku a management sítě v tabulce *switch*.

`switch_mining.pl`
Plní tabulku switch daty získané z aktivních prvků na základě dat vložených ze souboru nebo z příkazové řádky. Umožňuje manipulaci s řádky v tabulce.

`cname.pl`
Zajišťuje operace na tabulkou *pass*: vkládaní, mazaní a zobrazení záznamů.

`trap.pl`
Zajišťuje zpracování zpráv typu alert, schéma je na obrázku 6.7.

`set.pl`
Vzdáleně ovládá aktivní prvky.

`find.sh`
Zobrazí výsledek hledaní MAC adresy, IP adresy a portu na kterém je umístěna. Zobrazí aktivitu portu určitého prvku.

`get_vlan.sh`
Zobrazí id VLAN a interní id routeru, pro naplnění tabulky *vlan*.

7 Závěr

Cílem této diplomové práce byla realizace systému pro sledování rozsáhlé počítačové sítě. Bylo provedeno vyhodnocení náročnosti sběru dat v rozsáhlé počítačové síti se stovkami prvků a vytvořen uzel pro sběr a optimalizované ukládání dat z aktivních prvků sítě. Sběr dat se zabýval informacemi o konfiguraci aktivních prvků, aktuálním stavu a statistických provozu jednotlivých portů. Bylo vytvořeno centrální pracoviště pro zobrazování historií získaných dat, vyhodnování potenciálních problémů a vzdáleného ovládání aktivních prvků. Vývoj a realizace probíhala ve studentské síti KolejNet.

Vzhledem k tomu, že hardware všech serverů KolejNetu je na platformě PC z důvodu dobrého poměru cena/výkon. Bylo rozhodnuto použít tuto platformu pro sledování. Byl vybrán značkový server DELL, který po provedených testech lze označit, jako dostatečně výkonný.

Obsahuje redundatní zdroje napájení, výkonný řadič SAS disků s podporou RAID a dvě síťová rozhraní. Byl kladen důraz na ochranu serveru před přepětím v datové i napájecí části sítě, protože se často vyskytuje, jako důsledek atmosférických poruch. Server je možné podle potřeby rozšířit o pevné disky a operační paměť.

Server je umístěn v racku spolu se záložním zdrojem napájení. Je připojen do běžné a managovací sítě pomocí dvou síťových rozhraní, které zajišťují bezpečnost i výkon systému.

Při volbě operačního systému byl kladen důraz na snadnou obsluhu, správu serveru, údržbu a zabezpečení dat. Byl proto zvolen a použit OS FreeBSD, který v základní instalaci obsahuje většinu potřebných programů s kterými mají správci řadu let dobré zkušenosti. Pro sběr dat z aktivních prvků byla navržena vlastní databáze, jejíž návrh je dostatečně obecný a bude možné v budoucnu ho použít, jako zdroj informací pro IS.

V databázi jsou udržovány informace o aktivních prvcích a jejich stavu. Mezi informace, které jsou uchovány v databázi patří IP a MAC adresy, které jsou v síti používany. Z těchto informací lze zjistit která MAC adresa používá danou IP adresu a na kterém prvku a portu je do sítě připojena. Dále se ukládají informace o statistikách provozu na jednotlivých portech prvků a varovná hlášení, která informují o závažných událostech.

Existuje zde prostor pro další rozšíření, hlavně při přechodu na IPv6 se kterým se v následujících dvou letech počítá.

Slovníček pojmů a zkratek

ARP
– Address Resolution Protocol, 26

CAM
– Content Addressable Memory table , 25
CD
– Compact Disc – kompaktní disk, 41
CMIP/CMIS
– Common Management Information Protocol/Services, 23

datagram
– paket při nespolehlivém přenosu dat - UDP, 15
DBI
– perlové rozhraní pro přístup k databázím, 65
DDoS
– Distributed Denial of Service - distribuované útoky vedoucí k odmítnutí služby, 29
DHCP
– Dynamic Host Configuration Protocol, 38
DoS
– Denial of Service - útoky vedoucí k odmítnutí služby, 29
DVD
– "Digital Versatile Disk" nebo "Digital Video Disc" – formát optického nosiče, 41

FDB
– Forwarding DataBase – Virtuální LAN, 25
FDD
– Floppy Disk Drive) - jednotka pružných disků, 43

HDD
– Hard Disk Drive) - jednotka pevný disků, zjednodušeně pevný disk, 41

ICMP
– Internet Control Message Protocol – protokol řídících hlášení, 17
ISO/OSI
– standardizace počítačových sítí nazvané OSI organizaci ISO, 21

LAN
– Local Area Network – lokální síť, 13

man
– manuálové stránky pod OS unixového typu, 47
MIB
– Management Information Base – řídící informační báze, 24
MM
– Multi Mode – mnohavidová vlákna, 16

Netflow
– síťový protokol firmy CISCO, 17

NIC
– Network Interface Card – síťový adaptér (karta) , 39
NMS
– Network Management System – systém managementu sítí, 21

OPIE
– One-time Passwords In Everything – správce jednorázových hesel, 46
OS
– operating system - operační systém, 29

paket
– blok přenášených informací počítačovou sítí, 48
PAM
– Pluggable Authentication Modules – mechanizmus pro integraci více nízkoúrovňových autentizačních schémat do API, 51

RMON
– Remote Monitoring – vzdálené monitorování, 17

SAS
– Serial Attached SCSI, 41
SCSI
– Small Computer System Interface – komunikační rozhraní a sada příkazů pro výměnu dat mezi perifeií a sběrnicí, 41
sFlow
– standard pro sledování počítačových sítí, 17
SM
– Single Mode – jednovidová vlákna, 16
SMON
– Switch Monitoring – monitoring switchů , 17
SNMP
- Simple Network Management Protocol, 23
Simple Network Management Protocol, 17

TTL
– Time to live – parametr životnosti přenášeného paketu, 33

UNIX
původně Unics, podle Unary Information and Computing Service
- tvůrci Ken Thompson a Dennis Ritchi roku 1969 v Bell Laboratories, 43
UPS
– Uninterruptible Power Supply (Source) – nepřerušitelný zdroj energie, 41
USB
– Universal Serial Bus - univerzální sériová sběrnice, 41

VLAN
– Virtual Local Area Network, 25

WAN
– Wide Area Network – rozsáhlá síť, 13
WDM
– Wavelength Division Multiplex – vlnový multiplex, 16

8 Literatura

[1] Pravidla provozu počítačové sítě KolejNet. [online]. [cit. 7.07.2008].
Dostupné z: <http://www/docs/rules/sk-35-2001.html>

[2] Prohřeškový řád sítě KolejNet [online]. [cit. 7.07.2008].
Dostupné z:<http://www/docs/rules/prohreskovy_rad.html>

[3] Předpisy a nařízení [online]. [cit. 7.07.2008].
Dostupné z:<http://www/docs/rules/>

[4] Rada správců sítí [online]. [cit. 7.07.2008].
Dostupné z:<https://www.net.vutbr.cz/intra/rss>

[5] Freebsd.org [online].
Dostupné z:<http://www.freebsd.org/>

[6] Wikipedia-cz [online].
Dostupné z:<http://www.freebsdsoftware.org>

[7] Freebsd.org [online].
Dostupné z:<http://www.freebsd.org/>

[8] 3Com [online].
Dostupné z:<http://www.3com.com/>

[9] Manuály k přepínačům HP [online].
Dostupné z:<ftp://ftp.hp.com/pub/networking/software/>

[10] Freebsdsoftware.org [online].
Dostupné z:<http://www.freebsdsoftware.org/benchmarks/raidtest.html>

[11] CESNET - Performance Testing Tools [online].
Dostupné z:<http://www.cesnet.cz/doc/techzpravy/2003/perftools/>

[12] Dokumentace Net-SNMP [online].
Dostupné z:<http://net-snmp.sourceforge.net/>

[13] RFC1157 - Simple Network Management Protocol.
Dostupné z:<http://www.faqs.org/rfcs/rfc1157.html>

[14] CMIP/CMIS - Object Oriented Network Management [online].
Dostupné z:<http://www.cellsoft.de/telecom/cmip.htm/>

[15] CPAN [online].
Dostupné z: <http://search.cpan.org/>

[16] RMON[online].
Dostupné z:
<http://www.cisco.com/en/US/docs/internetworking/technology/handbook>

[17] CPAN -SNMP 5 [online].
Dostupné z:<http://search.cpan.org/ hardaker/SNMP-5.0401/SNMP.pm>

[18] Wikipedia-cz [online].
Dostupné z:<http://cs.wikipedia.org/>

[19] redteam-np.net [online].
Dostupné z:<http://www.redteam-np.net/>

[20] Lupa [online].
Dostupné z:<http://www.lupa.cz/>

[21] Lucas, M. *FreeBSD - Podrobný průvodce.*Brno: Computer Press, 2003.
ISBN 80-7226-795-7

[22] Pužmanová, R. *TCP/IP - V kostce.*České Budějovice: Koop, 2004.
ISBN 80-7232-236-2

[23] MOMJIAN, B. *PostgreSQL - Praktický průvodce.* Brno: Computer Press, 2003.
ISBN 80-7226-954-2

[24] SATRAPA P. *Perl pro zelenáče.* Praha: Neocortex, 2000. ISBN: 80-86330-02-8